PURDUE UNIVERSITY
SCHOOL OF ELECTRICAL ENGINEERING

INVESTIGATION OF OPTIMIZATION
OF ATTITUDE CONTROL SYSTEMS

Volume I

R. SRIDHAR, G. C. AGARWAL, R. M. BURNS,
D. M. DETCHMENDY, E. H. KOPF JR., R. MUKUNDAN

JANUARY, 1965

GPO PRICE $ _____

OTS PRICE(S) $ _____

Hard copy (HC) _____

Microfiche (MF) _____

Lafayette, Indiana

TR-EE65-3

Jet Propulsion Laboratory Contract 950670/SUBNAS 7-100

PRF-3807-55-285

INVESTIGATION OF OPTIMIZATION

OF ATTITUDE CONTROL SYSTEMS

Volume I

Authors:
R. Sridhar
G. C. Agarwal
R. M. Burns
D. M. Detchmendy
E. H. Kopf Jr.
R. Mukundan

School of Electrical Engineering

Purdue University

Lafayette, Indiana

January, 1965

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

LIST OF ILLUSTRATIONS

iv

LIST OF ILLUSTRATIONS (Continued)

LIST OF ILLUSTRATIONS (Continued)

CHAPTER 1

INTRODUCTION

This report presents the work done and the results

obtained during the year February 1964 - January 1965.

Certain parts of the material contained in this report have

been presented in the three quarterly reports submitted

previously. Repetition of this material was deemed desir-

able, however, in the interest of making this report a

presentation of the totality of the work done on the research

contract up to the present time, complete in and of itself,

with no necessity for referring to the previous quarterly

reports.

Each chapter is intended to be a complete presentation

of its own material, with no cross-referencing among the

individual chapters. In the interest of clarity and contin-

uity of presentation, however, some duplication of material

will be noted in various chapters, notably in chapters 3 and

4, in both of which is presented explanations of some of the

classical methods for the solution of the optimal control

problem.

Chapter 2 contains the development of a mathematical model for a space vehicle. A computer program suitable for use in simulation of the space vehicle on a digital computer is explained, and examples of the use of the program are given.

A review of optimal control theory is given in chapter 3. Some of the "classical" techniques for the solution of the optimal control problem are given, and their limitations are pointed out. The Specific Optimal Control approach to the optimal control solution is then presented, and several methods are given for finding the specific optimal solutions. It is felt that this approach to the solution of the attitude control problem may prove to be quite fruitful.

The optimal control problem in which the control input is bounded is examined in chapter 4. Shortcomings of the classical methods for solution of this problem are pointed out, and some methods of solution which overcome some of these difficulties are explained. Examples of the use of these methods are provided and comparisons of the methods are given.

Chapter 5 is concerned with sequential estimation of states and parameters in non-linear systems. A technique is developed with which sequential, least-square estimates

of the states of a system may be obtained, based on noisy
measurements of possibly nonlinear combinations of the states
of the system. The use of these estimates for the purpose
of controlling the system is investigated. Experimental
results of the use of the techniques developed in this chap-
ter are given.

In chapter 6, an additional approach to the solution of
the specific optimal control problem is given. This approach
makes use of a min-max criterion for the optimization. Such
a criterion requires that the maximum deviation be minimized,
as opposed to a least-squares criterion, in which a sum of
the squares of deviations is minimized. Examples are given
which illustrate the use of this approach in a specific
optimal control problem.

Appendices G, H, 1, J, and K contain listings of the
Fortran programs which were used to obtain the results given
in the various examples in this report. Explanations of
the functions of the programs are given in each appendix.
These appendices are in volume II.

CHAPTER 2

MATHEMATICAL MODEL AND DIGITAL COMPUTER

SIMULATION OF SPACE VEHICLES

2.1 Summary

The differential equations describing the motion of an

arbitrary space vehicle about its center of mass are deter-

mined and arranged in a form suitable for digital computer

solutions. From these equations a computer program is devel-

oped which allows the simulation of a space vehicle and its

attitude control system on an IBM 7094 or similar machine.

A test of the program is shown in Figure 2.1 where one of the

computer model's angular velocities is compared with tele-

metered values from Ranger VII's initial sun acquisition.

This computer model is now being used to investigate

the feasibility of new control schemes.

2.2 Equations of Motion

A mathematical model for a spacecraft is determined by

representing the vehicle as an invariant inertia tensor

written about a set of mutually perpendicular axes through

the center of mass. Any translational motion of the center

5
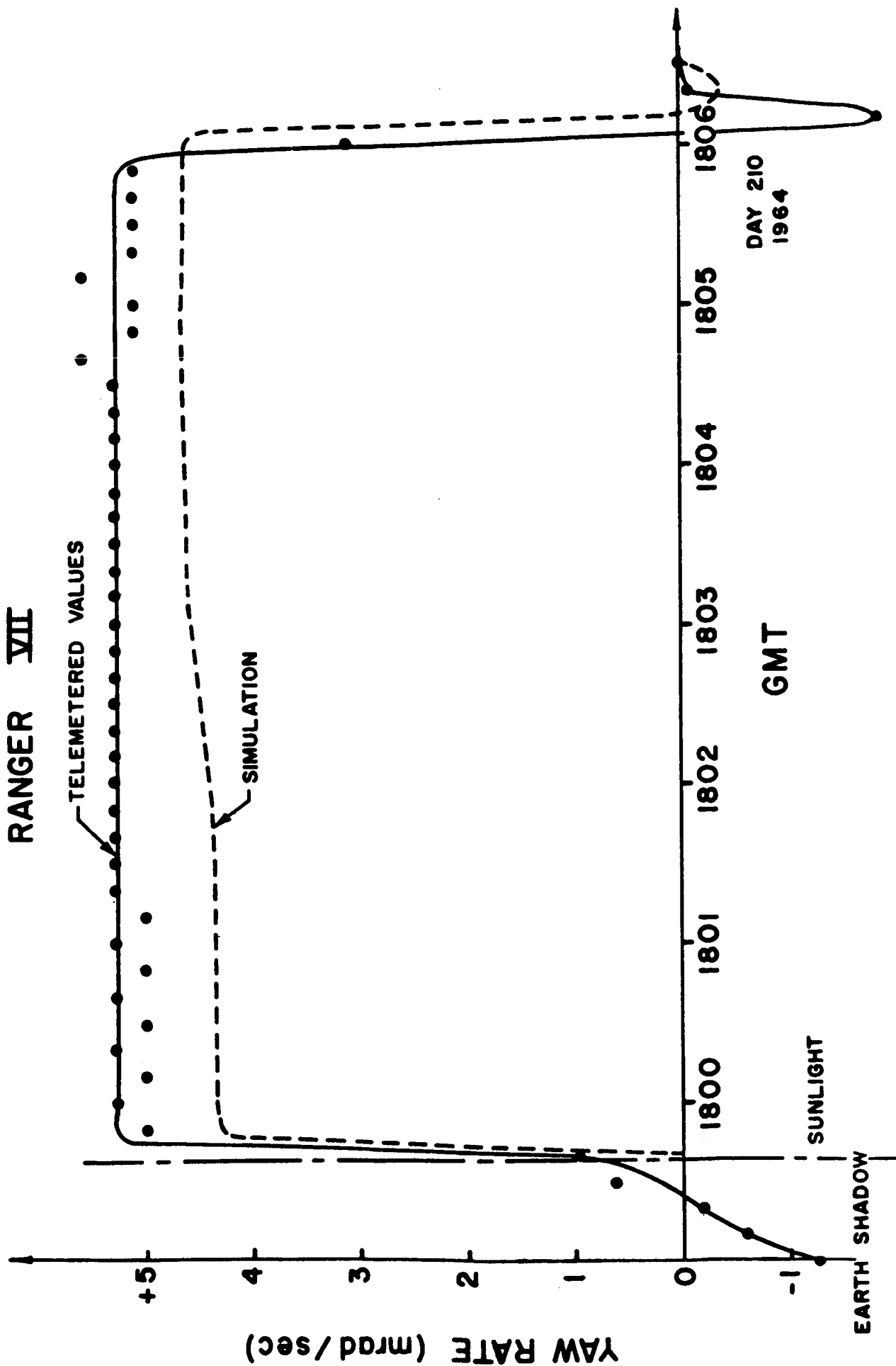
Figure 2.1

of mass is not considered because it would not affect the attitude control problem.

One criticism which might be raised at this point is that the moments of inertia are not constant throughout the flight of a spacecraft, but vary because of antenna angle changes and other factors. A time varying inertia tensor is not allowed, however, because it is not intended to simulate the entire attitude history of a vehicle with one computer run. What is intended is the simulation of certain portions of the flight, such as sun acquisition, where the inertia tensor can be considered constant.

2.2.1 Coordinate Systems

An inertially fixed cartesian coordinate system through the center of mass is assumed and represented by upper case letters X, Y and Z. In normal usage the Z axis is considered to point towards the sun. The vehicle control axes, called the body axes, are next represented in this inertial system and denoted by the lower case letters x, y, and z. See below and Figure 2.2.

SPACECRAFT
SHOWING BODY AXES
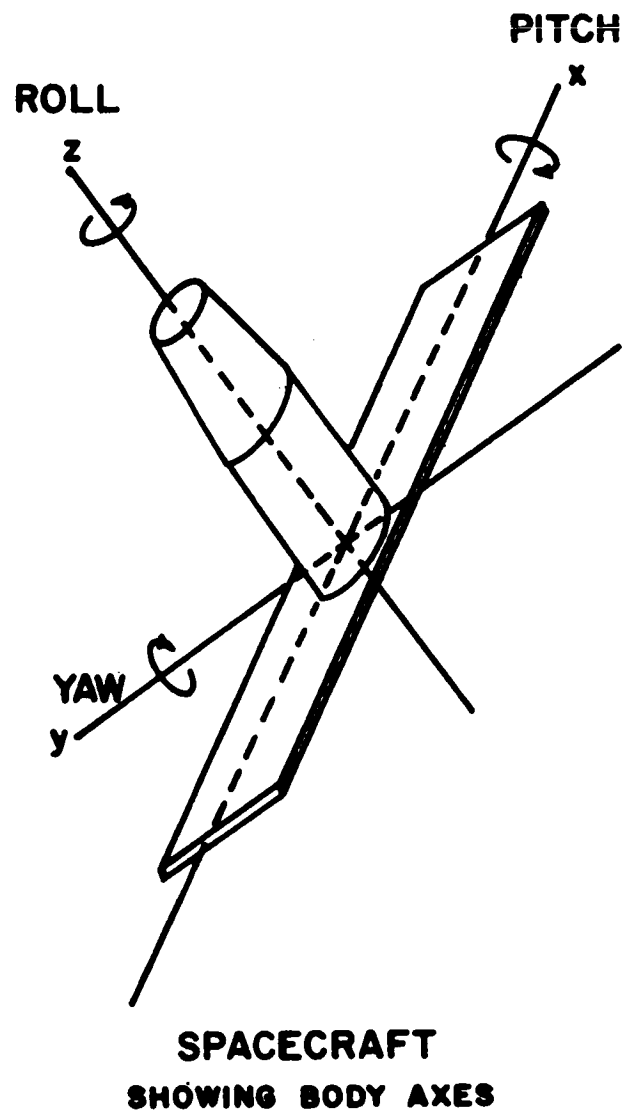
Figure 2.2
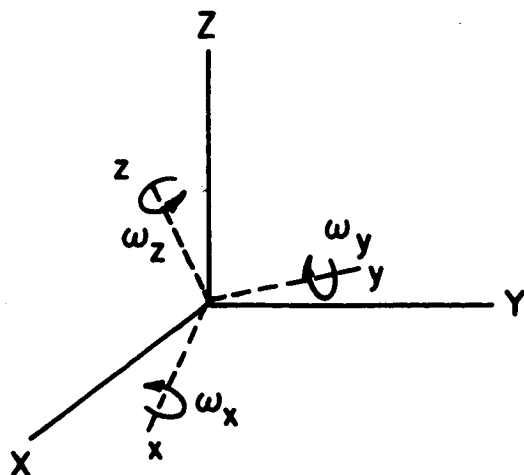
2.2.2 Description of the Angular Velocities of The Vehicle

Following the development in Goldstein [1] the angular momentum of a spacecraft is given by:

$$\underline{L} = I \underline{\omega} \qquad\qquad (2.2.1)$$

where I is the inertia tensor written about the control axes:

$$I = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{xy} & I_{yy} & I_{yz} \\ I_{xz} & I_{yz} & I_{zz} \end{bmatrix}$$

$\underline{\omega}$ is the angular velocity vector with components ω_x, ω_y, and ω_z about the three body axes.

$$\underline{\omega} = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}$$

The basic equation of motion is then:

$$\frac{d\underline{L}}{dt} = \underline{N} \qquad \text{where:} \qquad (2.2.2)$$

$$\underline{N} = \begin{bmatrix} N_x \\ N_y \\ N_z \end{bmatrix} \qquad \text{is the applied torque.}$$

It is understood here that the derivative is taken with respect to an inertially fixed system. Since the body system is rotating in inertial space, this derivative can be expressed as:

$$\left.\frac{d\underline{L}}{dt}\right|_{\text{inertial}} = \left.\frac{d\underline{L}}{dt}\right|_{\text{body}} + \underline{\omega} \times \underline{L} \qquad (2.2.3)$$

$$\left.\frac{d}{dt}\right|_{\text{body}} (I\,\underline{\omega}) = \underline{N} - \underline{\omega} \times I\,\underline{\omega} \qquad (2.2.4)$$

Assuming an invariant inertia tensor

$$I\,\underline{\dot{\omega}} = \underline{N} - \underline{\omega} \times \underline{I}\,\underline{\omega} \qquad (2.2.5)$$

Solving this linear algebraic system for $\underline{\dot{\omega}}$

$$\dot{\omega}_x = \frac{Det \begin{bmatrix} (N_x - \omega_y L_z + \omega_z L_y) & I_{xy} & I_{xz} \\ (N_y - \omega_z L_x + \omega_x L_z) & I_{yy} & I_{yz} \\ (N_z - \omega_x L_y + \omega_y L_x) & I_{zy} & I_{zz} \end{bmatrix}}{Det\ [I]} \tag{2.2.6}$$

$$\dot{\omega}_y = \frac{Det \begin{bmatrix} I_{xx} & (N_x - \omega_y L_z + \omega_z L_y) & I_{xz} \\ I_{yx} & (N_y - \omega_z L_x + \omega_x L_z) & I_{yz} \\ I_{zx} & (N_z - \omega_x L_y + \omega_y L_x) & I_{zz} \end{bmatrix}}{Det\ [I]} \tag{2.2.7}$$

$$\dot{\omega}_z = \frac{Det \begin{bmatrix} I_{xx} & I_{xy} & (N_x - \omega_y L_z + \omega_z L_y) \\ I_{yx} & I_{yy} & (N_y - \omega_z L_x + \omega_x L_z) \\ I_{zx} & I_{zy} & (N_z - \omega_x L_y + \omega_y L_x) \end{bmatrix}}{Det\ [I]} \tag{2.2.8}$$

2.2.3 Position Description by Euler Angles

Once an inertial reference is established, the body system rotation with respect to this reference may be described by three Euler angles. Consider a roll-pitch-roll

11

sequence as shown in Figure 2.3.

The first rotation yields a transformation

$$x' = x \cos \varphi + y \sin \varphi$$

$$y' = -x \sin \varphi + y \cos \varphi$$

$$z' = z$$

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = A' \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad ; \quad A' = \begin{bmatrix} \cos \varphi & \sin \varphi & 0 \\ -\sin \varphi & \cos \varphi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The second rotation yields a transformation:

$$x'' = x'$$

$$y'' = y' \cos \theta + z' \sin \theta$$

$$z'' = -y' \sin \theta + z' \cos \theta$$

$$\begin{bmatrix} x'' \\ y'' \\ z'' \end{bmatrix} = A'' \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} \quad ; \quad A'' = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & \sin \theta \\ 0 & -\sin \theta & \cos \theta \end{bmatrix}$$

The third rotation yields a transformation

$$x''' = x'' \cos \Psi + y'' \sin \Psi$$

$$y''' = -x'' \sin \Psi + y'' \cos \Psi$$

$$z''' = z''$$

1st ROTATION (ROLL)

2nd ROTATION (PITCH)

3rd ROTATION (ROLL)
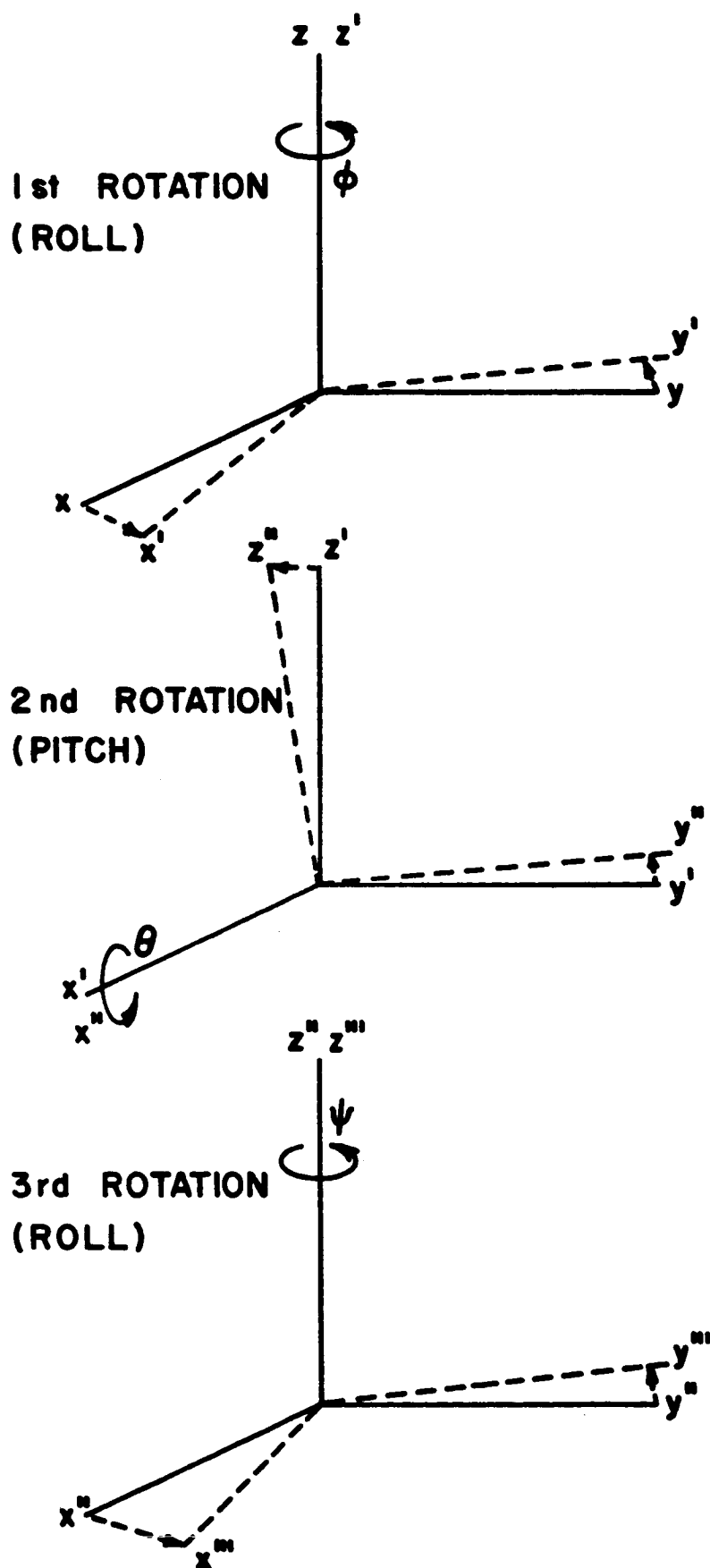
Figure 2.3

$$
\begin{bmatrix} x''' \\ y''' \\ z''' \end{bmatrix} = A''' \begin{bmatrix} x'' \\ y'' \\ z'' \end{bmatrix} \quad ; \quad A''' = \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}
$$

The third rotation fixes the final position of the body axes and hence the triple-prime coordinates may be identified as the body system. The transformation from the inertial system to the body system is then:

$$
\begin{bmatrix} x \\ y \\ z \end{bmatrix} = A''' A'' A' \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad \text{where:}
$$

$$
\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad \text{is in the inertial system and}
$$

$$
\begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad \text{is in the body system}
$$

What must now be determined is the relationship between the body rates and the rate of change of these Euler angles.

$\dot{\Psi}$ can be represented as a vector, using the right hand rule, along the z"' axis. This velocity then has components along each of the body axes:

$$\dot{\Psi}_x = 0$$

$$\dot{\Psi}_y = 0$$

$$\dot{\Psi}_z = \dot{\Psi}$$

In a similar fashion $\dot{\theta}$ is a vector along the x" axis and hence in the body system:

$$\begin{bmatrix} \dot{\theta}_x \\ \dot{\theta}_y \\ \dot{\theta}_z \end{bmatrix} = A"' \begin{bmatrix} \dot{\theta} \\ 0 \\ 0 \end{bmatrix}$$

$$\dot{\theta}_x = \dot{\theta} \cos \Psi$$

$$\dot{\theta}_y = -\dot{\theta} \sin \Psi$$

$$\dot{\theta}_z = 0$$

Finally for $\dot{\varphi}$:

$$\begin{bmatrix} \dot{\varphi}_x \\ \dot{\varphi}_y \\ \dot{\varphi}_z \end{bmatrix} = A"' A" \begin{bmatrix} 0 \\ 0 \\ \dot{\varphi} \end{bmatrix}$$

$$\dot{\varphi}_x = \dot{\varphi} \sin \Psi \sin \theta$$

$$\dot{\varphi}_y = \dot{\varphi} \cos \Psi \sin \theta$$

$$\dot{\varphi}_z = \dot{\varphi} \cos \theta$$

The sum of these rates must yield the body rates, $\underline{\omega}$:

$$\omega_x = \dot{\theta} \cos \Psi + \dot{\varphi} \sin \Psi \sin \theta$$

$$\omega_y = -\dot{\theta} \sin \Psi + \dot{\varphi} \cos \Psi \sin \theta$$

$$\omega_z = \dot{\Psi} + \dot{\varphi} \cos \theta$$

Solving for the Euler angle rates:

$$\dot{\varphi} = \frac{\omega_x \sin \Psi + \omega_y \cos \Psi}{\sin \theta} \qquad (2.2.9)$$

$$\dot{\theta} = \omega_x \cos \Psi - \omega_y \sin \Psi \qquad (2.2.10)$$

$$\dot{\Psi} = \omega_z - \dot{\varphi} \cos \theta \qquad (2.2.11)$$

2.3 Formulation of the Equations for Digital Solution

Equations (2.2.6)-(2.2.11) are six simultaneous first order nonlinear differential equations which describe completely the attitude of a space vehicle. Given a set of initial velocities and the initial position, if the torque $\underline{N}(t)$ is known these equations can be integrated by either an analog or digital computer to describe the motion as a

function of time.

Because of the many multiplications and trigonometric functions involved, it is essentially impossible to use an analog computer alone for this operation. This problem is, however, well within the scope of a large digital machine, such as an IBM 7094, when a numerical integration technique such as "Runge-Kutta Integration" is used.

2.3.1 Singularities

Before rushing ahead and writing a program to accomplish the solution of these equations, their nature should be investigated. It is noticed that the right hand sides of equations (2.2.9) and (2.2.11) possess singularities when the Euler angle θ attains the values:

$$\theta = \pm n\pi \quad ; \quad n = 0, 1, 2, \ldots$$

The temptation here is to rationalize in the following manner:

These singularities can be ignored because the computer can easily work with numbers 10^{+30} in magnitude and using floating-point numbers the chances of θ attaining a value such that $1/\sin \theta$ is greater than 10^{+30} are almost nil. Also the derivatives $\dot{\varphi}$ and $\dot{\psi}$ will probably not remain large for long and the resulting accuracy loss will be small.

To evaluate this assumption the physical interpretation of the singularities should be investigated. Consider a spacecraft aligned with the inertial system. The position description in a roll-pitch-roll sequence of Euler angles would be, of course, $\varphi = 0$, $\theta = 0$, and $\Psi = 0$. Now refer to Figure 2.4 and observe the sequence which describes the spacecraft after having made a yaw turn of $\Delta\alpha$ radians. The Euler angles are $\varphi = +\frac{\pi}{2}$, $\theta = +\Delta\alpha$, and $\Psi = -\frac{\pi}{2}$.

Since the original position with $\theta = 0$ was at one of the singularities, it can be seen that these singularities cannot be "glossed over". The magnitude of the changes in φ and Ψ when going through a rate singularity can be large, $\pm\frac{\pi}{2}$ in this example, and thus the position error accrued by numerically integrating through a singularity will be very large.

The technique used in the actual model is essentially the one proposed by P. Eckman [2]. Equations (2.2.9)-(2.2.11) may be written in terms of a different Euler sequence. If this new sequence is properly selected the rate singularities of the new Euler angles will not occur at the same _physical_ position of the body axes with respect to the inertial frame.

Consider a roll, pitch, yaw sequence where $\varphi' = $ roll,
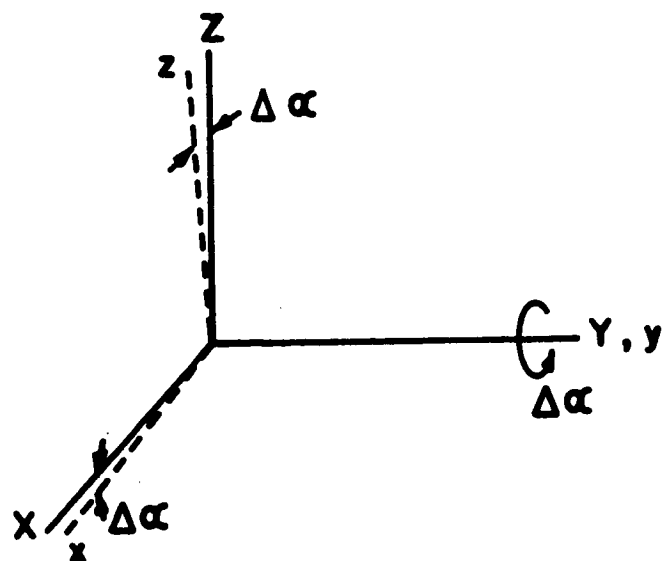
SPACECRAFT AFTER A $\Delta\alpha$ YAW TURN
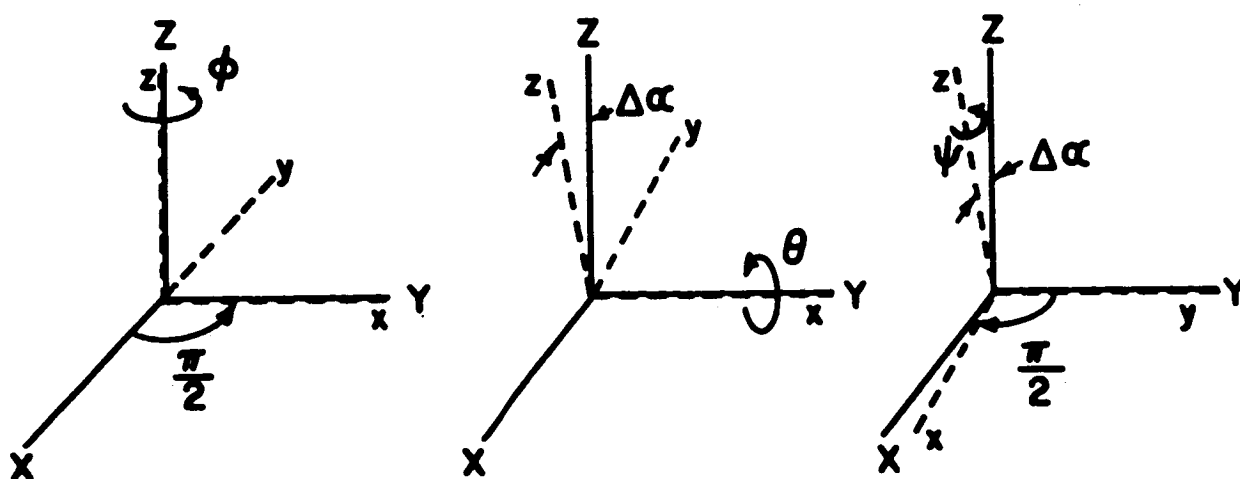
A ROLL-PITCH-ROLL EULER SEQUENCE FOR A

$\Delta\alpha$ YAW TURN

Figure 2.4

θ' = pitch, and ψ' = yaw. The new rate equations are:

$$\dot{\varphi}' = \frac{(\omega_z \cos \psi' - \omega_x \sin \psi')}{\cos \theta'} \qquad (2.3.1)$$

$$\dot{\theta}' = \omega_z \sin \psi' + \omega_x \cos \psi' \qquad (2.3.2)$$

$$\dot{\psi}' = \omega_y - \dot{\varphi}' \sin \theta' \qquad (2.3.3)$$

The technique used in the model is to integrate using one sequence of Euler angles until a rate singularity is approached. At this point the model switches Euler sequences and continues on using the new set of Euler angles.

The only problem here is that in order to switch sequences the present position must be determined using the new Euler sequence. The way in which this is accomplished is to note that the matrix A in the relationship:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = A \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

is invariant regardless of which set of Euler angles paramatrize it. Thus to change sequences A is computed in terms of the present sequence of Euler angles and then the inverse

operation is performed; that is, the new angles are found from the A matrix values.

2.3.2 Computer Program

The computer program which has been completed is completely described in a forthcoming Jet Propulsion Laboratory report from JPL Section 344. In brief the program is set up to numerically integrate using Runge-Kutta and Adams-Moulton techniques. The user of the program supplies the initial Euler angles and angular velocities along with the vehicle's inertia tensor. When these parameters and conditions are known, the user must write a subprogram to simulate the control system to be used. Completing this, the subprogram and data are fed to the computer along with the model program.

The computer will now produce a trajectory, printing out the Euler angles and angular velocities at prescribed intervals.

2.4 Control Subprograms

Two control subprograms have been developed for use with the model program. The first is for simulation of sun acquisitions using sun sensors and gyro rate feedback. The second is for the simulation of one cruise phase of a flight

using derived rate feedback and sun sensors.

2.4.1 Sun Sensor Model

The sun sensors (pitch and yaw) were modelled in the
following manner. It was assumed that the conventional
sun sensors would be used with solar cells and shadow masks.
This type of system has an "on axis" characteristic as
shown below. This is for one pitch sensor when the sun is
in the y-z plane.

V = SATURATED VOLTAGE LEVEL
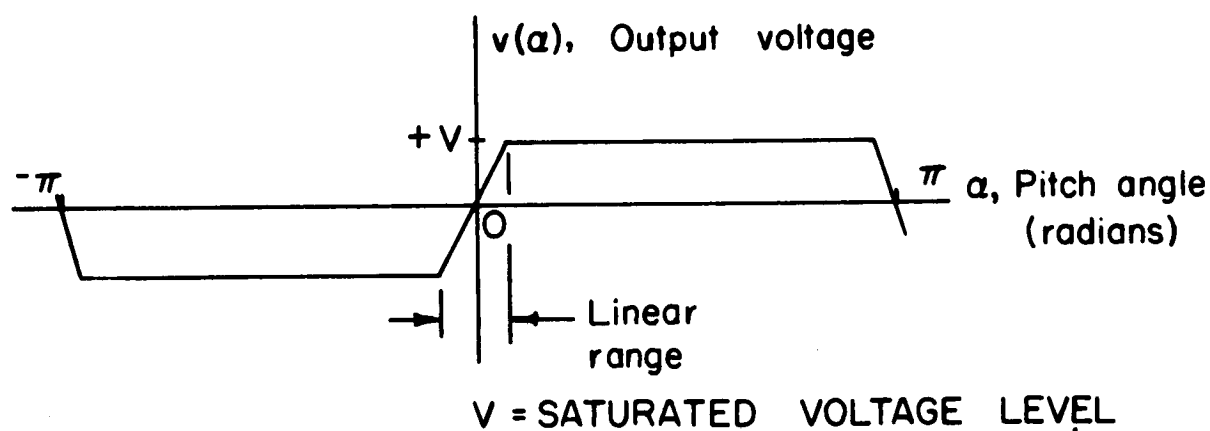
Now when the vehicle is positioned so that the sun is
no longer in the y-z plane the pitch output will be reduced
because the illuminated area of the cells decreases as the
cosine of the angle of offset from the x-y plane. This offset

angle is angle β for the pitch sensor and angle δ for the yaw sensor. See Figure 2.5.

The resulting sun sensor models produce outputs of $v(\alpha)$ cos β for pitch and $v(\gamma)$ cos δ for yaw. In the actual program the saturated voltage output and linear range are treated as input data.

2.4.2 Subprogram for Attitude Control During Sun Acquisition

This subprogram operates in the following manner. The numerical integration in the main program must have available the torques on the vehicle. When torque values are needed, the control subprogram is called. In this particular sub-program the present values of the Euler angles are sent to a routine simulating the sun sensors. Angles α, β, γ, and δ are calculated and the voltage outputs of the pitch and yaw sun sensors are determined. The angular velocities are then sent to a gyro-simulating routine where the proper scale factor is determined and the gyro output voltages are produced. Then the gyro voltages are summed with the sun sensor voltages and these sums are sent to the switching amplifier routine where the torques are determined.

SUN SENSOR ANGLES

DETAIL OF YAW
SENSOR ANGLES
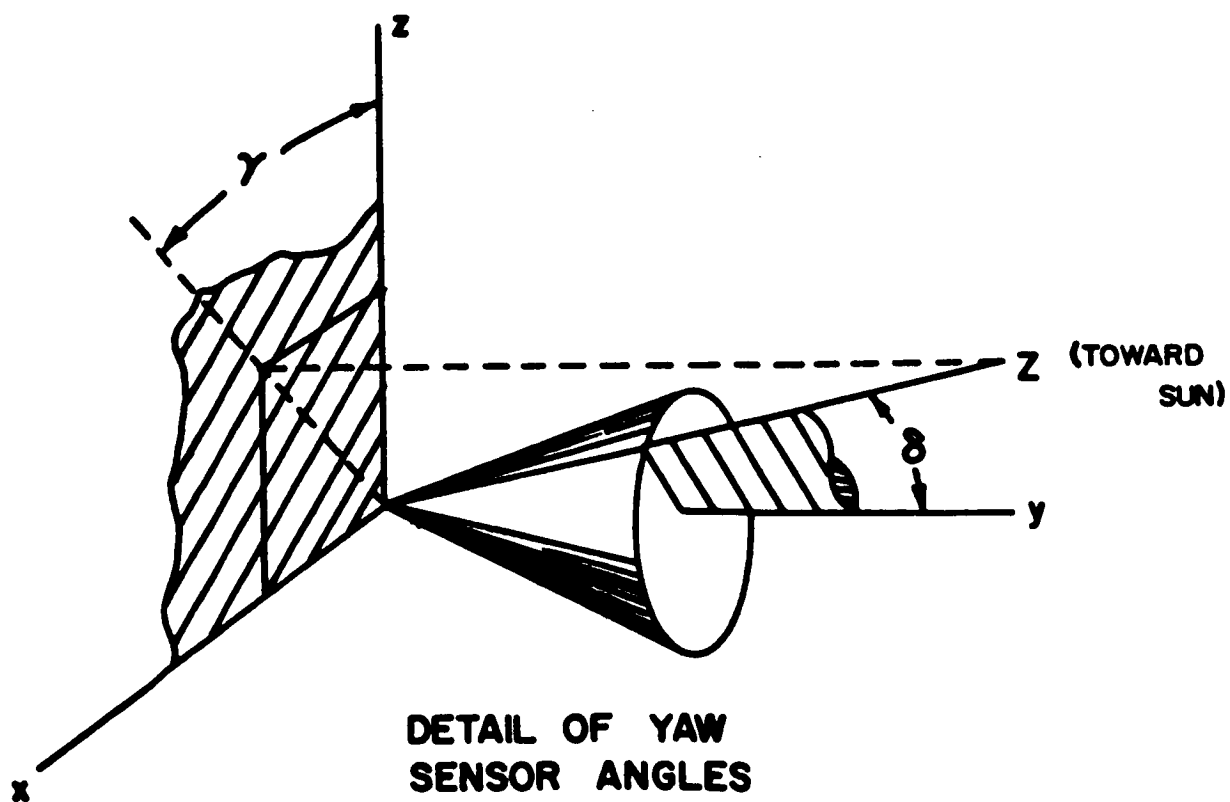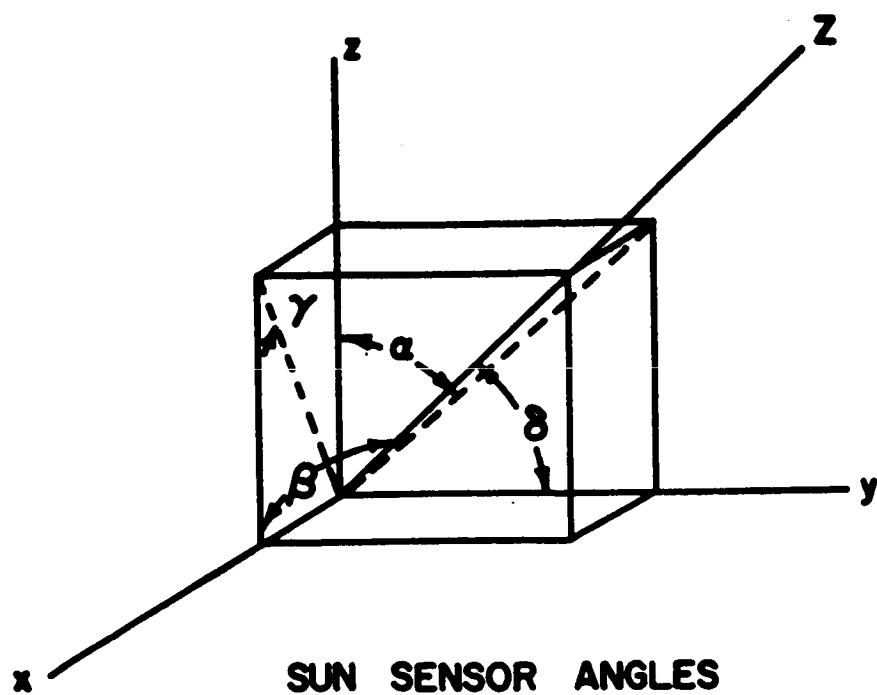
Figure 2.5

2.4.3 Derived Rate Cruise Mode Control Subprogram

The subprogram for derived rate control is similar to the acquisition program in that it uses identical sun sensor and switching amplifier routines. The major differences are: (i) constant solar torques are applied to the vehicle, (ii) the gyroscope routine is replaced by a derived rate routine, and (iii) a celestial sensor routine is added.

The performance of this control subprogram when incorporated into the overall model program can be seen from Figure 2.6. Here a Ranger-type vehicle was started with zero initial conditions under the influence of constant magnitude solar torques in pitch and yaw. The solar torques were made about an order of magnitude greater than those encountered in actual flights in order to conserve computer time. As can be seen from the figure, the system quickly established limit cycle operation in pitch. Yaw and roll rates are not shown, but limit cycle operation was also present in yaw and there was some roll motion due to the non-zero products of inertia. 2.88 volts was set as the switching level in the pitch error channel and from the figure the derived rate voltage increment at switching can easily be observed.

The derived rate feedback voltage is determined in the

RANGER VEHICLE
CRUISE MODE USING
DERIVED RATE
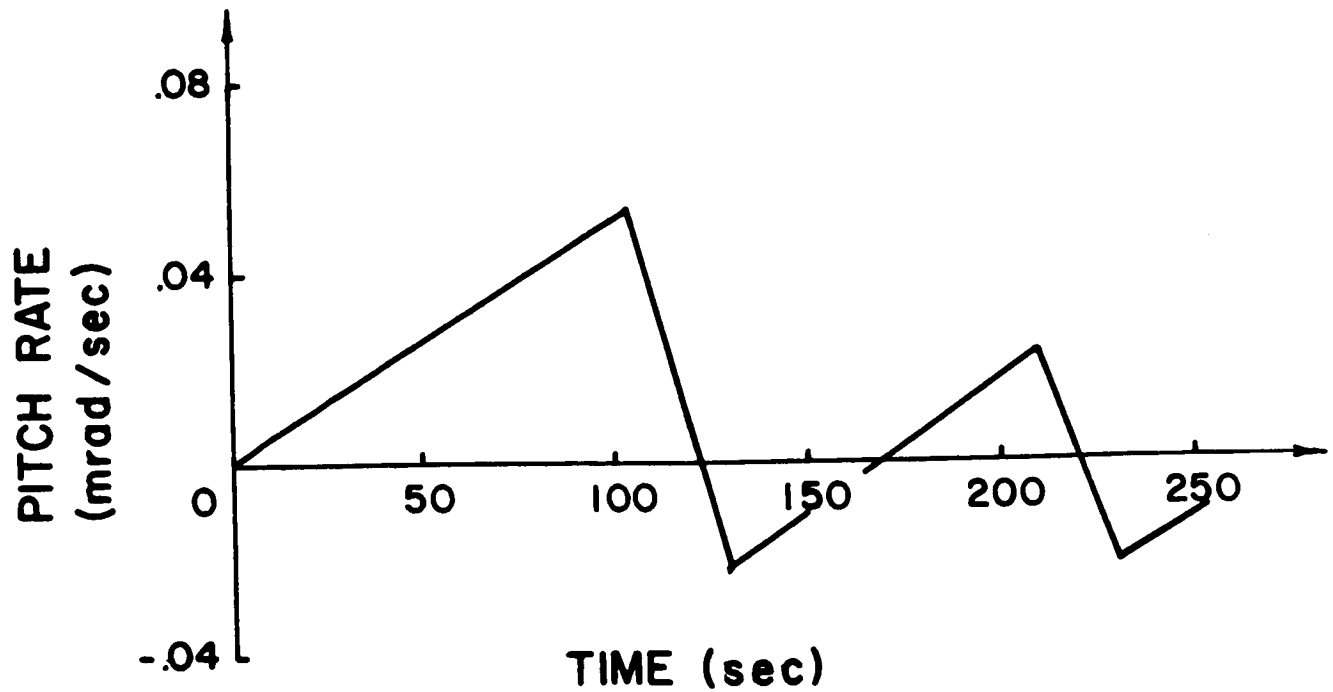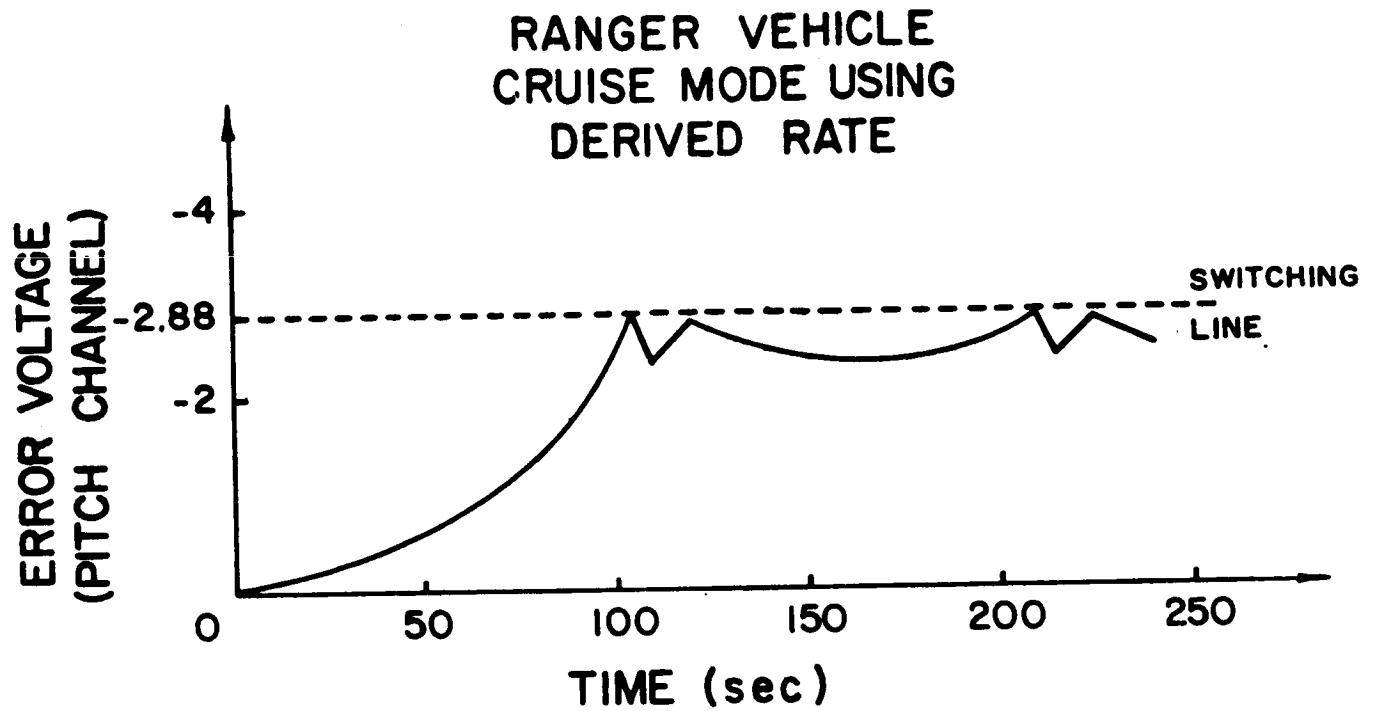
Figure 2.6

following manner. The controller subroutine is first called by the main program to determine the vehicle torques. This subprogram computes the pitch and yaw sun sensor outputs and the celestial sensor output. Next a derived rate subroutine is called which computes the proper time constants for charge or discharge and simple Euler integration is then used to determine the outputs from the derived rate networks. A "minimum-on time" is included by setting the main program for Runge-Kutta integration and forcing it to integrate with the torque applied for the minimum time.

2.5 Evaluation of the Program

Since a general analytical solution for the vehicle equations in not known it is somewhat difficult to check nemerical accuracy. Two types of checks have been made for a vehicle with a 0.6 mrad/sec^2 acceleration constant about all axes. These checks indicate at least 4 significant figure accuracy for rates and 3 significant figure accuracy for Euler angles over a 500 second (vehicle time) period when the integration step size range was set so that the computer running time (7094) was 2.5 minutes.

The first check was to set the initial roll and yaw rates to zero and apply torque only about the pitch axis.

This gives rise to an equivalent single degree of freedom

system, the response of which can be determined analytically.

The second check was to start with initial rates about all

axes and let the vehicle tumble with no applied torque. Since

the rate equations are norm-invariant for a principal-axis

system the sum of the squares of the rates weighted by their

respective moments of inertia should remain constant

2.6 Future Work

The simulation programs are included in this report in

Appendix G. As can be seen the routines are very segmented;

that is, every program is made up of many subroutines, all

having a uniform common area. This was done so that each

subroutine could be used separately or in various groups.

This feature will enable inclusion of the vehicle dynamics

in larger optimum control determining programs which hope-

fully will lead to a better understanding of optimization,

specific controllers, and the optimum solution for various

performance indices.

CHAPTER 3

THE OPTIMAL CONTROL PROBLEM

3.1 Summary

In this chapter, the possibility of using optimal control theory for solving a complex space age automatic control problem is examined. Such a problem for example, is the attitude control of a space vehicle. It appears that considerable modification of the theory is necessary before practical controllers can be devised for satisfactory operation of the space vehicle during the acquisition mode. The difficulties encountered in using classical optimal control theory are brought out. One can see that it is desirable to reformulate the problem as a specific optimal control problem. Several computational techniques for solving such specific optimal control problems are explained using many examples.

3.2 A Typical Optimal Control Problem

A typical optimal control problem is the following: The object to be controlled (the space vehicle) is described by a vector differential equation of the form

29

$$\dot{\underline{x}} \;=\; \underline{f}(t,\,\underline{x},\,\underline{u}) \qquad\qquad (3.2.1)$$

where $\underline{x}$ is a n-dimensional vector $(x_1,\, x_2,\, \ldots\, x_n)'$, the state of the system; and $\underline{u}$ is an m-dimensional vector $(u_1,\, u_2,\, \ldots,\, u_m)'$, the control vector. The prime denotes the transpose. The components $u_i(t)$, $i = 1,\, 2,\, \ldots,\, m$, are called the control functions. $\underline{f}$ is an n-dimensional vector $(f_1,\, f_2,\, \ldots,\, f_n)'$. The f_i, $i = 1,\, \ldots,\, n$, are assumed to possess piecewise continuous second partial derivatives with respect to all their arguments.

The control functions may be either unconstrained or may be required to fall within an allowable range of values. The general constraint on $\underline{u}(t)$ will be symbolically denoted by $\underline{u} \,\epsilon\, \Omega$ where Ω is a suitably defined set which in general is assumed to be closed. In most applications the $u_i(t)$, $i = 1,\, 2,\, \ldots,\, m$ are required to be at least piecewise continuous.

Let the object be in an initial state

$$\underline{x}(t_o) \;=\; \underline{c} \qquad\qquad (3.2.2)$$

The control problem is to find $\underline{u}(t)$ such that a given functional of $\underline{x}(t)$ and $\underline{u}(t)$, called the index of performance

or return function, of the form

$$I(\underline{u}) = \int_{t_0}^{T} g(t, \underline{x}(t), \underline{u}(t)) \, dt \qquad (3.2.3)$$

is minimized.

In equation (3.2.3) g is a scalar-valued function of its arguments and is assumed to possess piecewise continuous second partial derivations with respect to all its arguments. The terminal time T is assumed to be fixed in this discussion. In general it need not be so.

3.3 Classical Methods of Solution

Four classical methods that are available to solve this problem are (i) the Euler-Lagrange differential equations, (ii) Pantryagin's maximum principle, (iii) Bellman's dynamic programming, and (iv) Hamilton-Jacobi theory. No detailed derivations of these will be given; only the results will be stated. A brief "engineering" demonstration of these methods is given in Appendix A.

i. Euler-Lagrange differential equations

This method yields $\underline{u}^*(t)$ the optimal open loop solution. The method is as follows. Form the Lagrangian:

31

$$L(t, \underline{x}, \underline{u}, \underline{\lambda}) = g(t, \underline{x}, \underline{u}) + \langle \underline{\lambda}, \underline{f}(t, \underline{x}, \underline{u}) - \underline{\dot{x}} \rangle \qquad (3.3.1)$$

where $\underline{\lambda}$ is an n-dimensional multiplier vector. The Euler-Lagrange differential equations are given by:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \underline{\dot{x}}}\right) = \frac{\partial L}{\partial \underline{x}} \quad \rightarrow \quad -\underline{\dot{\lambda}} = \frac{\partial g}{\partial \underline{x}} + \frac{\partial f}{\partial \underline{x}} \underline{\lambda} \qquad (3.3.2)$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \underline{\dot{\lambda}}}\right) = \frac{\partial L}{\partial \underline{\lambda}} \quad \rightarrow \quad \underline{\dot{x}} = \underline{f}(t, \underline{x}, \underline{u}) \qquad (3.3.3)$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \underline{\dot{u}}}\right) = \frac{\partial L}{\partial \underline{u}} \quad \rightarrow \quad \frac{\partial g}{\partial \underline{u}} = \frac{\partial f}{\partial \underline{u}} \underline{\lambda} \qquad (3.3.4)$$

(equations (A.1.1), (A.4.10) and (A.4.11) in Appendix A)

Equations (3.3.2) and (3.3.3) are 2n ordinary differential equations with 2n boundary conditions given by the initial conditions $\underline{x}(t_o) = \underline{c}$ and the transversality conditions which are in this case $\underline{\lambda}(T) = \underline{0}$. Equation (3.3.4) is a finite equation which yields $\underline{u}$ as a function of $\underline{x}(t)$ and $\underline{\lambda}(t)$. This is used to eliminate $\underline{u}$ in equations (3.3.2) and (3.3.3), and the resulting two point boundary value problem is solved for $\underline{x}^*(t)$ and $\underline{\lambda}^*(t)$. Substituting for $\underline{x}^*(t)$ and $\underline{\lambda}^*(t)$ in (3.3.4) results in $\underline{u} = \underline{u}^*(t)$.

The Euler-Lagrange method as outlined here implicitly assumes that the components of the control vector $\underline{u}$ in equation (3.2.1) are unconstrained. This is certainly not the case in the space vehicle attitude control problem. The method can be modified to take care of bounded control. This modification, in effect, leads to the use of the Pontryagin's maximum principle which is discussed next.

The Euler-Lagrange differential equations are necessary but not sufficient conditions for an optimal solution.

ii. <u>Pontryagin's Maximum Principle</u>.

Form the Hamiltonian; $H(t, \underline{x}, \underline{\lambda}, \underline{u})$ defined as

$$H(t, \underline{x}, \underline{\lambda}, \underline{u}) = \langle \underline{\lambda}, \underline{f}(t, \underline{x}, \underline{u}) \rangle + g(t, \underline{x}, \underline{u}) \qquad (3.3.5)$$

where $\underline{\lambda}$ is an n-dimensional multiplier vector. $\underline{u} = \underline{u}^*(t, \underline{x}, \underline{\lambda})$ is obtained by minimizing H with respect to $\underline{u}$ alone. Set:

$$\frac{\partial H}{\partial \underline{u}} \bigg|_{\underline{u} = \underline{u}^*} = 0 \qquad (3.3.6)$$

(Note: equation (3.3.6) is true only if the minimum occurs interior to the set of admissable values for $\underline{u}$. In general the minimization of the Hamiltonian is performed over the admissible range of the u's.) Define:

$$H^*(t, \underline{x}, \underline{\lambda}) = \underset{\underline{u}(t) \, \epsilon \, \Omega}{\text{Min}} \quad H(t, \underline{x}, \underline{\lambda}, \underline{u}) \tag{3.3.7}$$

This minimization will yield $\underline{u}^*$ explicitly (at least in principle) in the form

$$\underline{u}^* = \underline{u}^*(t, \underline{x}, \underline{\lambda}) \tag{3.3.8}$$

Thus

$$H^*(t, \underline{x}, \underline{\lambda}) = H(t, \underline{x}, \underline{\lambda}, u^*(t, \underline{x}, \underline{\lambda})) \tag{3.3.9}$$

Form the canonic equations:

$$\dot{x}_i = \frac{\partial H^*(t, \underline{x}, \underline{\lambda})}{\partial \lambda_i} \qquad i = 1, \ldots\ldots\ldots, n \tag{3.3.10}$$

$$\dot{\lambda}_i = - \frac{\partial H^*(t, \underline{x}, \underline{\lambda})}{\partial x_i} \qquad i = 1, \ldots\ldots\ldots, n \tag{3.3.11}$$

(equations (A.4.18) and (A.4.19) in Appendix A)

The solution of equations (3.3.10) and (3.3.11) subject to the intial conditions $\underline{x}(t_o) = \underline{C}$ and the transversality conditions yields the optimal trajectory $\underline{x}^*(t)$ and $\underline{\lambda}^*(t)$. This solution is substituted into $\underline{u}^* = \underline{u}^*(t, \underline{x}, \underline{\lambda})$ to yield the optimal control function.

In general, the solution of the Euler-Lagrange equa-
tions or the canonic equations yields the optimum open loop
solution. The Hamilton-Jacobi or dynamic programming formu-
lation of the optimum control problem will yield the closed
loop or "feedback law" solution. These methods are dis-
cussed next.

iii. Bellman's dynamic programming.

Dynamic programming is a powerful tool that can be
used to solve, in principle, a variety of multi-stage deci-
sion processes. This notion is made clear if one considers
the duration of the process, $(T-t_o)$, to be divided into a
finite number of time intervals. The problem then is to
choose a control vector $\underline{u}$ as a function of the state $\underline{x}$ at
the beginning of each of these time intervals such that the
performance index attains a minimum value. It is clear that
this will lead to an optimal control law.

The functional equation of dynamic programming which is
often referred to in the literature as the Bellman equation
is derived in Appendix A.

Since the minimum value of the performance index de-
pends on the initial state $\underline{C}$ and the starting instant τ,
define the "return function" or "value function" $J(\underline{C}, \tau)$ as

$$J(\underline{C},\ \tau) = \underset{\substack{\underline{u}(t)\ \epsilon\ \Omega \\ \tau\ \leq\ t\ \leq\ T}}{\text{Min}} \int_{\tau}^{T} g(t,\ \underline{x},\ \underline{u})\ dt \qquad (3.3.12)$$

subject to the differential constraint (3.2.1) for a process starting at time τ with the initial state $\underline{C}$ and terminating at fixed time T.

The return function satisfies the equation

$$J(\underline{C},\ \tau) = \underset{\underline{u}(t)\ \epsilon\ \Omega}{\text{Min}} \left[g(\tau,\ \underline{C},\ \underline{u}(\tau))\ \Delta\ +\ J(\underline{C}\ +\ \underline{f}(\tau,\ \underline{C},\ \underline{u}(\tau))\Delta, \right.$$
$$\left. \tau\ +\ \Delta)\ +\ 0(\Delta^{2}) \right] \qquad (3.3.13)$$

Equation (3.3.13) is the discrete version of the Bellman equation which is useful for numerical solutions.

The continuous version of the Bellman equation is (equation (A.2.14) in Appendix A)

$$\frac{\partial J}{\partial t}\ +\ \underset{\underline{u}(t)\ \epsilon\ \Omega}{\text{Min}} \left[g(t,\ \underline{x}^{*},\ \underline{u}(t)) \right.$$
$$\left. +\ \langle \underline{f}(t,\ x^{*},\ \underline{u}(t)),\ \nabla_{x}\ J \rangle \right]\ =\ 0$$

$$(3.3.14)$$

The boundary condition on (3.3.14) is

$$J(\underline{x}^{*},\ T)\ =\ 0 \qquad (3.3.15)$$

In equation (3.3.14), the variables $\underline{C}$ and τ have been replaced by $\underline{x}^*$ and t, the "current state" on an optimal trajectory and current time respectively.

The solution of (3.3.14) with boundary condition (3.3.15) will yield the optimal control law in the form, $\underline{u}^* = \underline{u}^*(t, \underline{x}^*)$.

iv. Hamilton-Jacobi Theory

In equation (3.3.14) when the minimization is performed, the resulting equation is called the Hamilton-Jacobi partial differential equation.

From equation (3.3.5)

$$H(t, \underline{x}^*, \underline{u}, \underline{\lambda}^*) = g(t, \underline{x}^*, \underline{u}) + \langle \underline{f}(t, \underline{x}^*, \underline{u}), \underline{\lambda}^* \rangle$$

(3.3.16)

In terms of the minimum value of the Hamiltonian, equation (3.3.14) is equivalent to (equation (A.2.20) in Appendix A)

$$\frac{\partial J}{\partial t} + H^*(t, x^*, \nabla_{x^*}J) = 0 \qquad (3.3.17)$$

Equation (3.3.17) when solved with the boundary condition (3.3.15) will yield $J(\underline{x}^*, t)$. The multiplier vector $\underline{\lambda}^*$ is evaluated from the relation

$$\underline{\lambda}^* = \nabla_{x^*}J \qquad (3.3.18)$$

Thus

$$\underline{u}^*(t, \underline{x}^*) = \underline{u}^*(t, \underline{x}^*, \underline{\lambda}^*)\Big|_{\underline{\lambda}^* = \nabla_{x^*}J}$$

The optimal control law is explicitly obtained as a function of the current time and the state.

3.4 Critique of Classical Methods

The purpose of this section is to outline the difficulties encountered when the methods presented above are applied to control problems, with emphasis on application to the attitude control problem.

Consider first the Euler-Lagrange differential equations (3.3.2) through (3.3.4). For a given set of initial conditions on $\underline{x}$ these equations constitute a two point boundary value problem, i.e. conditions on $\underline{x}$ at the initial time and conditions on $\underline{\lambda}$ at the terminal time are specified. In general two point boundary value problems are difficult to solve.

One of the computational methods which appears to be promising for solving problems of this type is quasilinearization. An outline of this method is given in Appendix B.

After determining $\underline{x}(t)$ and $\underline{\lambda}(t)$ then $\underline{u}^*(t)$ the so-called control function or open loop control can be determined. Nothing at all is said concerning the synthesis problem, i.e. finding $\underline{u}^*(t, \underline{x})$. Since this is the problem of interest, the solution of the Euler-Lagrange equations provides only limited data.

The Maximum Principle provides a different theoretical approach to the optimization problem which is particularly useful for the case of bounded control. Practically, however, the resulting canonic equations to be solved, i.e. equations (3.3.10) and (3.3.11), represent the same type of problem as the Euler-Lagrange differential equations. Both sets of differential equations represent two point boundary value problems and in many cases the equations are identical. Hence the Maximum Principle and the Euler-Lagrange equations provide means of determining the control function $\underline{u}^*(t)$. There still remains the synthesis problem. It is the exceptional case where the control law can be determined by the above methods.

This leads then to the Hamilton-Jacobi approach, i.e. equation (3.3.17). The solution to this nonlinear partial differential equation will determine the control law, i.e.

39

$\underline{u} = \underline{u}^*(t, \underline{x})$. The difficulty is, of course, in solving the Hamilton-Jacobi equation which is a nonlinear partial differential equation. In general this is a formidable problem and there is no guarantee that the resulting control law can be implemented in a practical manner.

The Dynamic Programming approach, equation (3.3.13), provides a practical method of solving the Hamilton-Jacobi equation which at the same time preserves the physical characteristics of the problem and yields some insight. This technique provides a computational scheme for solving many optimization problems. When it is applied to a control problem of the type being considered the results of the computations would be tables of numbers which would specify the control $\underline{u}$ as a function of the state variables. A solution to the synthesis problem which could be instrumented directly is not provided.

Moreover there is an inherent difficulty which is far more serious than the ones outlined above. Briefly, the difficulty arises in that with the above mentioned methods it is necessary to assume that all of the state variables are available in order to attempt the synthesis problem. There is no theory available which would allow incorporating

constraints which specify which states are available to be
fed back.

It is this practical difficulty which restricts the
applicability of the above methods to the attitude control
problem. Thus a reformulation of the optimization problem,
which incorporates the physical constraints placed on the
attitude control problem, is necessary. This leads directly
to the problem of specific optimal control.

3.5 The Specific Optimal Control Problem

In many practical situations, even if an optimum con-
trol law can be synthesized, it will not be a satisfactory
solution because of the complexity of the dependence of
the optimum control law on the state of the system and on
the time.

Often, the form of dependence of the control law, not
necessarily optimum, on the state is known beforehand except
for a finite set of parameters. The known form depends on
the manipulations that are possible with the available phys-
ical equipment.

In the attitude control problem under investigation the
number of states available for measurement is restricted;
and also the reliability of the controller used to perform

the objectives of the mission is important.

In order to incorporate these factors, the problem will be formulated in the following fashion and will be termed the problem of Specific Optimal Control [3].

Problem Statement

The specific optimal control problem is defined in the following manner:

Given a plant with dynamic equation of the form

$$\dot{\underline{x}} = \underline{f}(\underline{x}, u) \qquad (3.5.1)$$

where $\underline{x}$ is an n-dimensional vector, the state of the system; u is a scalar, the control function; $\underline{f}$ is a n-dimensional vector. More generally, the control function can be an m-dimensional vector (i.e., the plant would be a multi-input plant) and $\underline{f}$ can be an explicit function of time t (i.e., the plant is time-varying).

Let the plant be in an initial state

$$\underline{x}(0) = \underline{c} \qquad (3.5.2)$$

Determine the unknown parameters in a control law of the form

$$u = h(\underline{y}, \underline{b}) \qquad (3.5.3)$$

where $\underline{y}$ is a p-dimensional vector which is a known function of the state $\underline{x}$ and $\underline{b}$ is a q-dimensional constant vector of the unknown parameters to be determined, such that an index of performance of the form

$$I_1(u) = \int_0^T g(\underline{x}, u) \, dt \qquad (3.5.4)$$

is minimized, where $g(\underline{x}, u)$ is a scalar valued function of its arguments and T is the fixed terminal time. More generally, g can be an explicit function of time t (i.e., the performance is weighted as a function of time).

The f_i, i = 1, ..., n, and g are assumed to possess piecewise continuous second partial derivatives with respect to all of their arguments.

3.6 Proposed Methods of Solution

The following methods are proposed for solving the specific optimal control (SOC) problem;

(i)   Parameter Optimization

(ii)   Transformation to two-point boundary-value problem (TPBVP)

(iii)   Differential Approximation.

All these methods are basically computational techniques and are equally applicable to both linear and nonlinear systems.

These techniques are explained below using several examples.

i. Parameter optimization

The SOC problem may be written as follows:

Given

$$\dot{\underline{x}} = \underline{F}(\underline{x}, \underline{b}) \qquad (3.6.1)$$

with $\underline{x}(0) = \underline{C}$, where

$$\underline{F}(\underline{x}, \underline{b}) = \underline{f}(\underline{x}, h(\underline{y}, \underline{b})) \qquad (3.6.2)$$

Determine the parameter vector $\underline{b}$ such that

$$I(\underline{b}) = \int_{0}^{T} G(\underline{x}, \underline{b}) \, dt \qquad (3.6.3)$$

is minimized.

In equation (3.6.3)

$$G(\underline{x}, \underline{b}) = g(\underline{x}, h(\underline{y}, \underline{b})) \qquad (3.6.4)$$

From equations (3.6.1) and (3.6.3), the SOC problem may be rewritten in the following way.

Given a fixed configuration system (3.6.1) with q parameters b_1, b_2, ..., b_q and an index of performance which is some continuous function $I(b_1, b_2, ..., b_q)$ of the variable parameters, i.e.,

$$Z = I(b_1, b_2, \ldots, b_q) \qquad (3.6.5)$$

Specify an algorithm for determining the arguments b_1, b_2,..,bq which will minimize the function I by observing the value Z_i where $Z_i = I(b_{1i}, b_{2i}, \ldots, b_{qi})$ for a sequence of parameter settings. The surface defined by (3.6.5) is called the IP surface (index of performance surface).

The problem of determining optimal search procedures for locating the absolute minimum (or maximum) of a function of variables is a difficult problem [4]. Even in the case where it is known, a priori, that the function is unimodal, the proboem has been resolved only for functions of one variable [5].

The systems mechanizing such algorithms to extremize the function I are called optimizers, automatic optimalizers, extremal control systems, or hill-climbers. The majority of the techniques proposed in the literature will work satis-factorily only if the function I has a single minimum (the relative minimum problem). Here, a simple modified gradient method is presented for the solution of the SOC problem [6]. For various other schemes see [7, 8].

Stepwise Version of Gradient Method

It is desired to move toward a minimum of I by correct-
ing a set of approximations to the values of the b_i which
make $\partial I/\partial b_i = 0$, $i = 1, 2, \ldots, q$. The corrections are made
by increments proportional to the negative of the gradient,
i.e., if $b_i^{(p)}$ is the pth approximation for b_i and $\partial I/\partial b_i$
is the gradient at this point with respect to b_i, then $(p+1)^{th}$
approximation is taken as (for example)

$$(3.6.6)$$

$$b_i^{(p+1)} = b_i^{(p)} - \frac{\partial I}{\partial b_i} \Delta \sigma_i, \quad i = 1, 2, \ldots, q$$

where $\Delta \sigma_i$ is a constant and is chosen depending on the
amount of correction desired at each step.

Assuming the function I has only one minimum, the n-
dimensional minimization problem can be reduced to a sequence
of one-dimensional minimization problems. The minimum in
the direction $i = j$ is obtained by taking the gradient with
respect to b_j and following the gradient until I reaches a
minimum. In many cases it is possible to obtain 3 points
such that the minimum lies inside the two extreme points and
then fit a parabolic curve through these points and find the
minimum value of this parabola. The parameter value that
yields the least value for I with the parabolic fit is taken

as one of the next trial points and the minimization is done
by approximating by another parabola whose minimum is deter-
mined. The procedure is repeated until the desired accuracy
is obtained. In the course of computations it is sometimes
necessary to modify the value of $\Delta\sigma$ as the minimum is ap-
proached.

Example 3.1

Consider a second order plant described by the differ-
ential equations

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = -2x_1 - 3x_2 + u \tag{3.6.7}$$

Let the initial conditions be

$$x_1(0) = C_1$$

$$x_2(0) = C_2 \tag{3.6.8}$$

and the index of performance be

$$I_1(u) = \int_0^T (x_1^2 + x_2^2 + u^2)\, dt \tag{3.6.9}$$

Let the desired controller be of the form

$$u = A x_1 + B x_2 \tag{3.6.10}$$

where A and B are unknown to be determined so as to minimize the index of performance $I_1(u)$.

Substitution of (3.6.10) into (3.6.7) and (3.6.9) gives:

$$\dot{x}_1 = x_2$$

$$ \text{(3.6.11)} $$

$$\dot{x}_2 = -2x_1 - 3x_2 + Ax_1 + Bx_2$$

and

$$I(A,B) = \underset{A,B}{\text{Min}} \int_0^T [x_1^2 + x_2^2 + (Ax_1 + Bx_2)^2] \, dt \quad \text{(3.6.12)}$$

For a numerical solution, let

$$C_1 = 2.0, \quad C_2 = 2.0 \quad \text{and} \quad T = 1.0$$

Let the initial approximation for A and B be

$$A = -0.2$$

$$B = -0.2$$

The computer results are as follows:

(i) Initial Approximation

   $A = -0.2, \quad B = -0.2, \quad I = 5.19875783$

(ii) Search for minimum in A direction

   $A = -0.052021 \qquad B = -0.2$

   $I = 5.11067343$

(iii) Search for minimum in B direction

A = -0.052021   B = -0.223471

I = 5.11922168

(iv) Search for minimum in A direction

A = -0.05521   B = -0.223471

I = 5.11018097

(v) Search for minimum in B direction

A = -0.05521   B = -0.227940

I = 5.11018044

(vi) Search for minimum in A direction

A = -0.05521   B = -0.227940

I = 5.11018044

(vii) Search for minimum in B direction

A = -0.05521   B = -0.226007

I = 5.11017662

Thus, the optimum values of the feedback coefficients are

A = -0.05521

B = -0.226007

and the minimum value of the index of performance is

I = 5.11017662.

For comparison, this specific optimal solution is compared with the open-loop optimal solution. The optimal solution is obtained as follows:

The plant equations are:

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = -2x_1 - 3x_2 + u$$

and the index of performance

$$I_1(u) = \frac{\text{Min}}{\substack{u(t) \\ 0 \leq t \leq T}} \int_0^T (x_1{}^2 + x_2{}^2 + u^2) \, dt$$

Define the Lagrangian as in (3.3.1)

$$L = (x_1{}^2 + x_2{}^2 + u^2) + \lambda_1(x_2 - \dot{x}_1) + \lambda_2(-2x_1 - 3x_2 + u - \dot{x}_2) \quad (3.6.13)$$

The Eular-Lagrange equations are

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = -2x_1 - 3x_2 + \lambda_2$$

$$\dot{\lambda}_1 = -2x_1 + 2\lambda_2$$

$$\dot{\lambda}_2 = -2x_2 + 3\lambda_2 - \lambda_1$$

$$u = -0.5\lambda_2$$

(3.6.14)

The last equation in (3.6.14) is an algebraic relation. The boundary conditions on (3.6.14) are

$$x_1(0) = C_1 = 2.0 \quad , \quad x_2(0) = C_2 = 2.0$$

$$\lambda_1(T) = \lambda_2(T) = 0 \quad , \quad T = 1.0$$

This two-point boundary-value problem (TPBVP) is readily solved by the quasi-linearization method [Appendix B]. The performance index for the open-loop optimal control is

$$I_1(u = u^*(t)) = 5.10841614$$

Notice that the index of performance for the specific optimal system in very close to the index of performance for the open-loop optimal system.

The specific optimal trajectories (x_1^S, x_2^S) and $u^S = (Ax_1^S + Bx_2^S)$ are compared with the open-loop optimal trajectories (x_1^*, x_2^*) and $u^* = u^*(t))$ and are shown in figure 3.1. It is interesting to note that the specific optimal trajectory matches the optimal trajectory very closely.

The Fortran II program for the IBM 7094 machine for parameter optimization is given in Appendix H.

ii <u>Transformation to two-point boundary-value problem (TPBVP)</u>

The parameter optimization method, presented above, for the type of problems represented by the equations (3.6.1) and

51
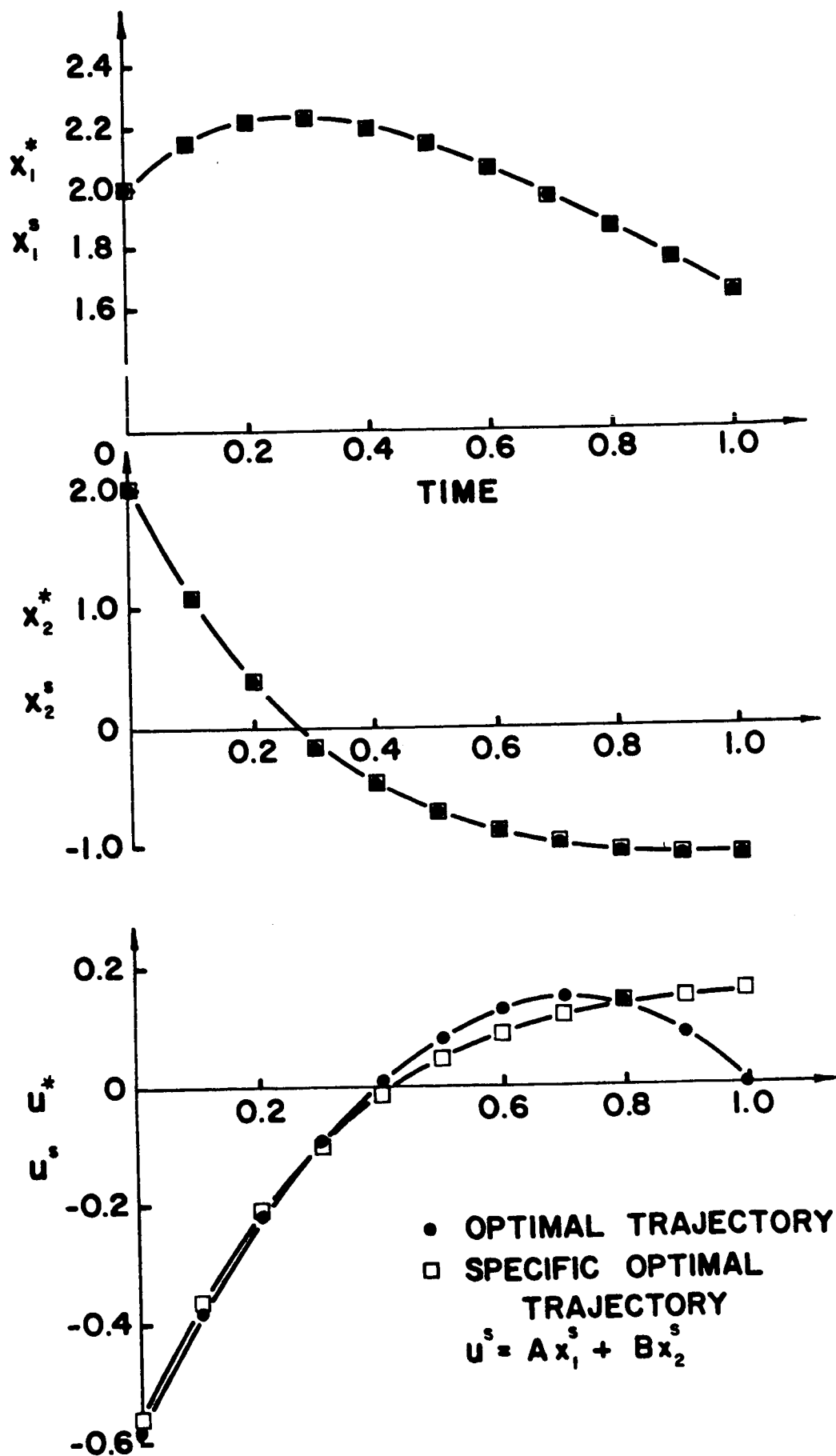
FIGURE 3.1   Specific Optimal Trajectory Compared with the Optimal

(3.6.3), i.e.,

$$\dot{\underline{x}} = \underline{F}(\underline{x}, \underline{b})$$

(3.6.1)

$$\underline{x}(0) = \underline{C}$$

and

$$I(\underline{b}) = \underset{\underline{b}}{\text{Min}} \int_0^T G(\underline{x}, \underline{b}) \, dt$$

(3.6.3)

is a suitable approach to the solution if the vector $\underline{b}$ is of low dimension. However, it is necessary that the boundary conditions on (3.6.1) be of the type $\underline{x}(0) = \underline{C}$, i.e., only initial conditions may be specified. If mixed boundary conditions are given on the equation (3.6.1), i.e., some at the initial point $t = 0$ and some at the terminal point $t = T$, then it is necessary to view the problem as a TPBVP.

The basic idea is to consider $\underline{b}$ to be a part of the state vector [9]

$$\underline{b} = \underline{b}(t)$$

(3.6.15)

Since $\underline{b}$ is a constant vector,

$$\dot{\underline{b}} = 0$$

(3.6.16)

By adjoining the equation (3.6.16) to the equation
(3.6.1) the specific optimal control problem is reduced to
an ordinary problem of minimization of an integral with
differential constraints. The unknown initial conditions
$\underline{b}(0)$ are determined in the course of solving the resulting
Euler-Lagrange equations subject to the given boundary con-
ditions on the equation (3.6.1) and certain other free
boundary conditions obtained from the transversality con-
dition [Appendix A].

The Lagrangian L is

$$L = G(\underline{x}, \underline{b}) + \langle \underline{\lambda}, F(\underline{x}, \underline{b}) - \dot{\underline{x}} \rangle + \langle \mu, -\dot{\underline{b}} \rangle \qquad (3.6.17)$$

where $\lambda(t)$ is an n-dimensional multiplier vector and $\mu(t)$
is a q-dimensional multiplier vector.

The Euler-Lagrange equations are

$$\dot{\underline{x}} = \underline{F}(\underline{x}, \underline{b})$$

$$\dot{\underline{b}} = 0 \qquad (3.6.18)$$

$$-\dot{\underline{\lambda}} = \frac{\partial G}{\partial \underline{x}} + \left(\frac{\partial \underline{F}}{\partial \underline{x}} \right)'$$

$$-\mu = \frac{\partial G}{\partial \underline{b}} + \left(\frac{\partial \underline{F}}{\partial \underline{b}} \right)' \underline{\lambda}$$

where

$$\frac{\partial G}{\partial \underline{x}} \triangleq \left(\frac{\partial G}{\partial x_1}, \frac{\partial G}{\partial x_2}, \ldots, \frac{\partial G}{\partial x_n} \right)'$$

$$\frac{\partial G}{\partial \underline{b}} \triangleq \left(\frac{\partial G}{\partial b_1}, \frac{\partial G}{\partial b_2}, \ldots, \frac{\partial G}{\partial b_q} \right)'$$

$$\frac{\partial \underline{F}}{\partial \underline{x}} \triangleq \begin{bmatrix} \frac{\partial F_1}{\partial x_1} & \cdots & \frac{\partial F_1}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial F_n}{\partial x_1} & \cdots & \frac{\partial F_n}{\partial x_n} \end{bmatrix} \quad (n \times n) \text{ matrix}$$

$$\frac{\partial \underline{F}}{\partial \underline{b}} \triangleq \begin{bmatrix} \frac{\partial F_1}{\partial b_1} & \cdots & \frac{\partial F_1}{\partial b_q} \\ \vdots & & \vdots \\ \frac{\partial F_n}{\partial b_1} & \cdots & \frac{\partial F_n}{\partial b_q} \end{bmatrix}, \quad (n \times q) \text{ matrix}$$

The natural boundary conditions for this problem are

(as obtained from the transversality conditions)

$$\underline{x}(0) = \underline{C} \qquad \underline{\lambda}(T) = 0$$

$$(3.6.19)$$

$$\underline{\mu}(0) = \underline{\mu}(T) = 0$$

The set of equations (3.6.18) represent $(2n + 2q)$ ordinary differential equations, nonlinear in general, with boundary conditions given by (3.6.19). There are various techniques available in the literature for the solution of TPBVP, e.g. "shooting" methods, gradient methods, and quasi-linearization [6, 10, 11]. The method of quasi-linearization seems very promising for the solution of a TPBVP and is explained in Appendix B. This method is relatively simple to program and has favorable convergence properties; in fact quadratic convergence is assured when suitable restrictions are placed on the TPBVP [10].

Example 3.2

Consider a second order nonlinear plant described by the equations

$$\dot{x} = y$$

$$(3.6.20)$$

$$\dot{y} = -(x^2-1)y - x + u$$

Let the initial conditions be

$$x(0) \ = \ c_1$$

<div align="right">(3.6.21)</div>

$$y(0) \ = \ c_2$$

The object is to find the specific optimal control law of the form

$$u \ = \ bx \qquad\qquad (3.6.22)$$

where b is the unknown constant and x is the only accessible state, such that the performance index

$$I_1 \ = \ \int_0^1 \ (x^2 + y^2 + u^2) \ dt \qquad\qquad (3.6.23)$$

is minimized.

Adjoin to (3.6.20) the differential equation

$$\dot{b} \ = \ 0 \qquad\qquad (3.6.24)$$

By eliminating u, the specific control problem is reduced to an ordinary problem of minimization of an integral with differential constraints. The resulting equations are

$$\dot{x} \ = \ y$$

$$\dot{y} \ = \ -(x^2-1)y - x + bx \qquad\qquad (3.6.25)$$

$$\dot{b} \ = \ 0$$

with the initial conditions

$$x(0) = C_1$$

$$y(0) = C_2$$

and the index of performance

$$I = \int_0^1 (x^2 + y^2 + b^2 x^2)\, dt \qquad (3.6.26)$$

The Lagrangian for the problem is

$$L = (x^2 + y^2 + b^2 x^2) + \lambda(y-\dot{x}) + \mu(-x^2 y + y - x + bx - \dot{y})$$

$$+ \xi(-\dot{b})$$

where λ, μ, and ξ are multipliers.

The Euler-Lagrange equations for the minimization problem are

$$\dot{x} = y$$

$$\dot{y} = -x^2 y + y - x + bx$$

$$\dot{b} = 0$$

$$\dot{\lambda} = -2x - 2b^2 x + 2\mu xy + \mu - b\mu \qquad (3.6.27)$$

$$\dot{\mu} = -2y - \lambda + \mu x^2 - \mu$$

$$\dot{\xi} = -2bx^2 - \mu x$$

The natural boundary conditions for the problem are

$$x(0) = C_1 \quad , \quad y(0) = C_2$$

$$\lambda(1) = \mu(1) = 0 \qquad\qquad (3.6.28)$$

$$\xi(0) = \xi(1) = 0$$

For application of the quasilinearization method, let the initial conditions on $b(t)$, $\lambda(t)$, and $\mu(t)$ be

$$b(0) = -0.05$$

$$\lambda(0) = 0 \qquad\qquad (3.6.29)$$

$$\mu(0) = 0$$

Then the initial approximation to $x(t)$, $y(t)$, $b(t)$, $\lambda(t)$, $\mu(t)$, and $\xi(t)$ is obtained by integrating the nonlinear differential equations (3.6.27) with initial conditions (3.6.28) and (3.6.29). (In general, this solution will not satisfy the terminal conditions in (3.6.28).)

The $(r+1)$-st approximation is determined from the r-th approximation via the relations

$$\dot{x}_{r+1} = y_{r+1}$$

$$\dot{y}_{r+1} = (-2x_r y_r - 1 + b_r)x_{r+1} + (-x_r^2 + 1)y_{r+1} + x_r b_{r+1}$$

$$\qquad\qquad + 2x_r^2 y_r - b_r x_r$$

$$\dot{b}_{r+1} = 0 \qquad\qquad (3.6.30)$$

$$\dot{\lambda}_{r+1} = (-2 - 2b_r^2 + 2\mu_r y_r)x_{r+1} + (2\mu_r x_r)y_{r+1}$$

$$- (4b_r x_r + \mu_r)b_{r+1} + (2x_r y_r + 1 - b_r)\mu_{r+1}$$

$$- 4x_r y_r \mu_r + 4b_r^2 x_r + \mu_r b_r$$

$$\dot{\mu}_{r+1} = (2\mu_r x_r)x_{r+1} - 2y_{r+1} - \lambda_{r+1} - (x_r^2 - 1)\mu_{r+1} - 2\mu_r x_r^2$$

$$\dot{\xi}_{r+1} = (-4b_r x_r - \mu_r)x_{r+1} - 2x_r^2 b_{r+1} - x_r \mu_{r+1} + 4b_r x_r^2$$

$$+ \mu_r x_r$$

The boundary conditions on (3.6.30) are

$$x_{r+1}(0) = C_1 \qquad\qquad y_{r+1}(0) = C_2$$

$$\lambda_{r+1}(0) = \mu_{r+1}(1) = 0 \qquad\qquad (3.6.31)$$

$$\xi_{r+1}(0) = \xi_{r+1}(1) = 0$$

The numerical solution of the linear system of equations
(3.6.30) is readily obtained by determining the homogeneous
and particular solutions and appropriately selecting the
constant multipliers for the homogeneous solutions, thereby

constructing a solution satisfying the boundary conditions (3.6.31).

For numerical solution, let the initial conditions on the system be

$$x(0) = C_1 = 1.0 \quad , \quad y(0) = C_2 = 1.0$$

Only 3 iterations are required for satisfactory convergence in this example. The value of b converges as follows:

Initial Approximation	$b = -0.05$
First iteration	$b = -0.135108$
Second iteration	$b = -0.135743$
Third iteration	$b = -0.135744$

The specific optimal trajectories (x^S, y^S) are shown in figure 3.2.

The feedback coefficient is

$$b = -0.13574$$

and the corresponding value of the index of performance is

$$I(u = bx) = 1.84932$$

Systems with time-varying deterministic inputs can be handled easily as illustrated by the following example.
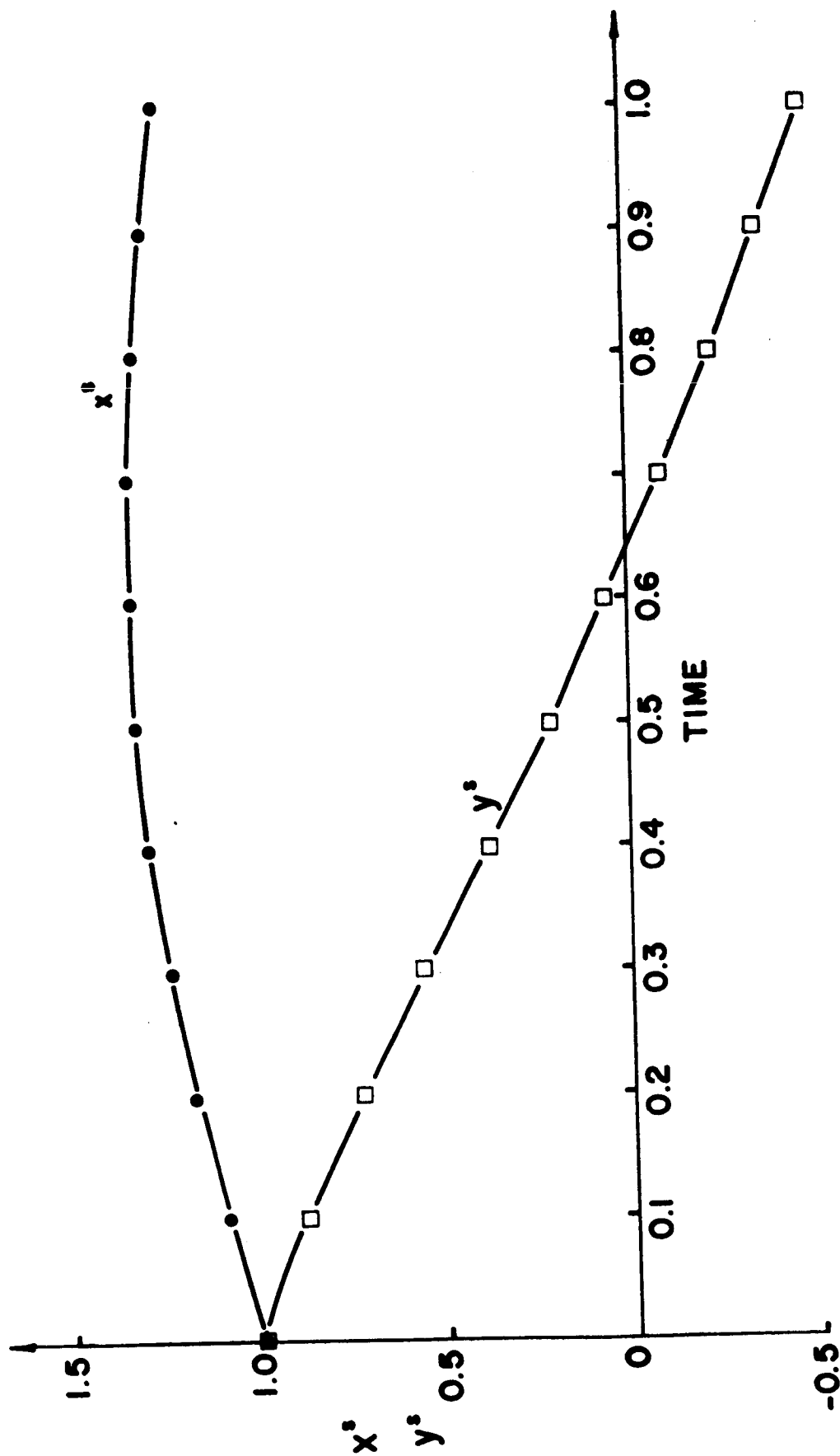
FIGURE 3.2   Specific Optimal Trajectory

<u>Example 3.3</u>:

Consider a plant described by the equation

$$\dot{x} = - x + \xi + u \qquad (3.6.22)$$

Let the system be subjected to external time-varying input (or disturbance) ξ of the form

$$\xi = 0.1 \ \text{Sin} \ 10t \qquad (3.6.33)$$

Let the plant be in an initial state

$$x(0) = C \qquad (3.6.34)$$

The object is to find the Specific Optimal Control law of the form

$$u = b \ x \qquad (3.6.35)$$

where b is the unknown constant such that the performance index

$$I_1 = \frac{1}{2} \int_0^T (x^2 + u^2) \ dt \qquad (3.6.36)$$

is minimized. Here, T is the fixed terminal time.

This problem is easily reduced to a boundary value problem as outlined in example 3.2. The results are as follows:

Terminal Time	Initial Condition	Feedback Coefficient
T	0	b
0.5	1.0	-0.21008
1.0	0.5	-0.33454
1.0	1.0	-0.33418
1.0	1.5	-0.33404

Figure 3.3 shows the specific optimal trajectory x^S, the external time-varying input ξ and the control $u = bx^S$ plus the input ξ for two-initial conditions $C = 1.5$ and $C = 0.5$ and the terminal time $T = 1.0$.

It is interesting to note that the specific optimal control law, in general, depends on the initial state of the system and the duration of the process. One is now forced to ask the question: How does the feedback coefficient b depend upon T and C? This is the so-called sensitivity problem and will be considered later.

iii Differential Approximation

In many situations, it is required to choose the best controller from a set of controllers. For example, if two state variables are available, say x and y, then one has to consider the several forms of controllers that are easy to
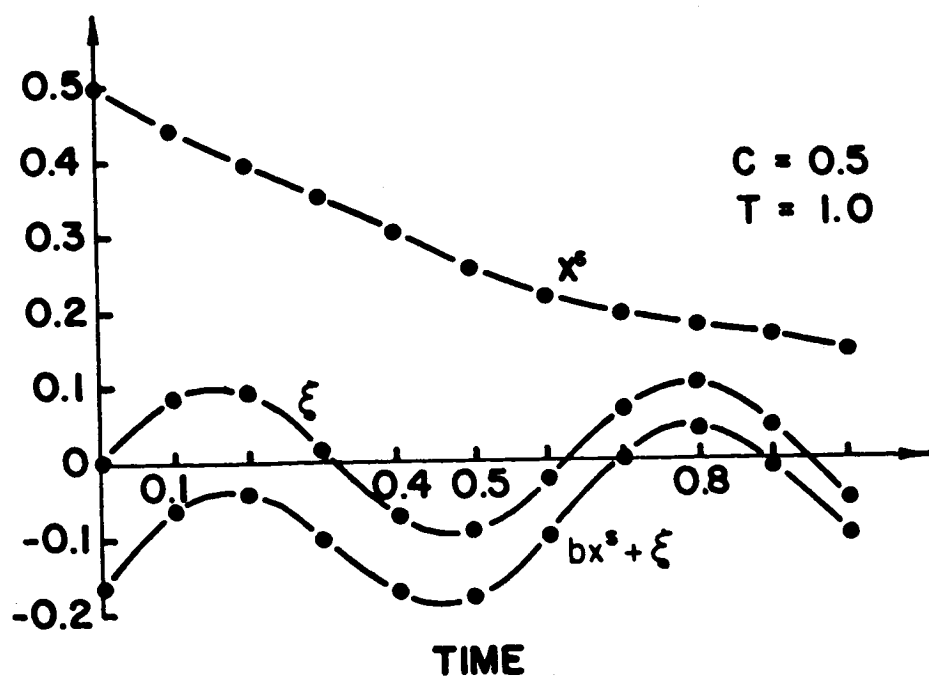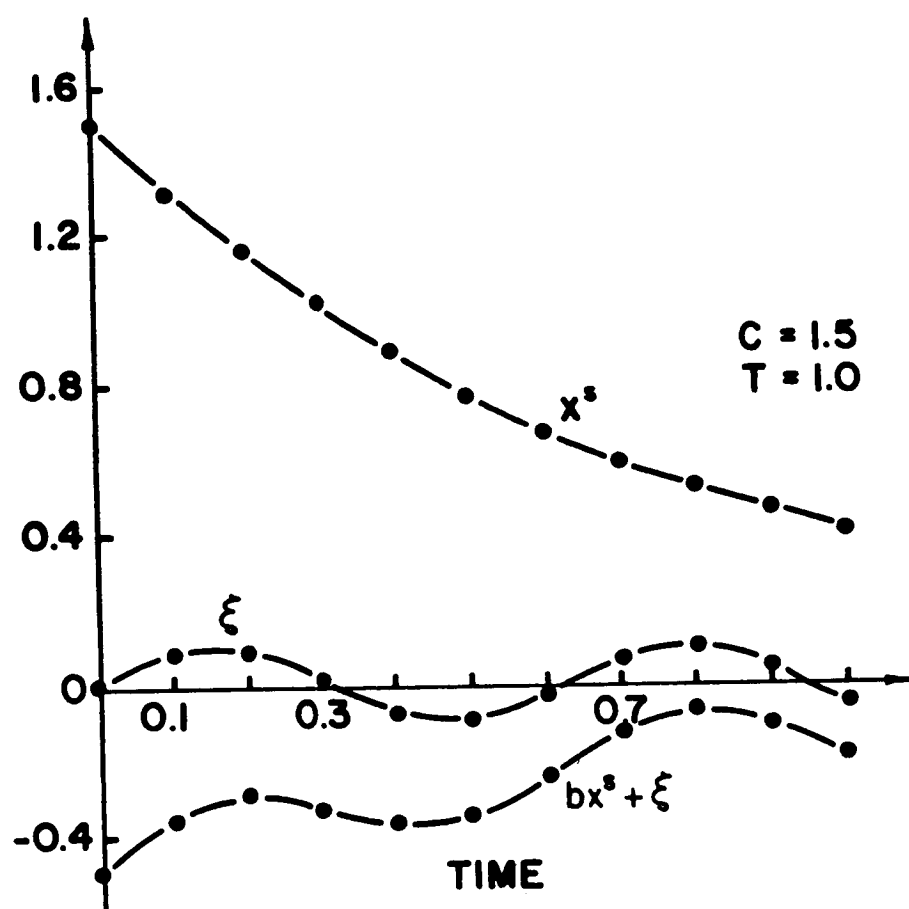
FIGURE 3.3 Specific Optimal Trajectory, System with

Time-varying Deterministic Input

instrument, e.g. $u = bx + cy$, $u = bx + dx^3 + cy$, $u = bx + cy + dy^3$, etc. The previous two methods, i.e., parameter optimization method and the transformation to TPBVP, may be used to obtain the specific optimal control laws of the given form and then select the best out of this set. However, the repeated application of these methods will require a considerable amount of programming effort and computer time. The method of differential approximation is particularly suitable in such cases.

Philosophically, this method is different from the previous methods. The solution here requires knowledge of the open-loop optimal solution ($u^* = u^*(t)$) in order to synthesize the closed loop solution. The open-loop optimal solution consists of the optimal control function $u^*(t)$ and the corresponding optimal trajectory $x^*(t)$ which do not depend on the form of the specific controller. In general, the solution using this method will result in a slight degradation of performance compared to the previous methods. However, the computations necessary with this method are often easier to perform [Appendix C].

Let the optimal trajectory without the specific controller constraint be $\underline{x}^*(t) = \varphi(t)$. Equation (3.6.1) represents the system equation with a specific controller. It is

easy to see that if there exists a set $\underline{b}$ such that

$$\dot{\underline{\varphi}}(t) - \underline{F}(\varphi(t), \underline{b}) = 0 \tag{3.6.37}$$

$$0 \leq t \leq T$$

then the set $\underline{b}$ is the optimal parameter set for the specific controller. However, in general, equation (3.6.37) will not be satisfied.

Therefore, one intuitively feels that an acceptable solution may be one which makes the left hand side of equation (3.6.37) "close to zero", the closeness being defined in a suitable manner. For example, $\underline{b}$ may be obtained as the solution of

$$\underset{\underline{b}}{\text{Min}} \int_0^T || \dot{\underline{\varphi}}(t) - \underline{F}(\underline{\varphi}(t), \underline{b}) ||^2 \, dt \tag{3.6.38}$$

or

$$\underset{\underline{b}}{\text{Min}} \underset{0 \leq t \leq T}{\text{Max}} || \dot{\underline{\varphi}}(t) - \underline{F}(\dot{\underline{\varphi}}(t), \underline{b}) || \tag{3.6.39}$$

where, in equations (3.6.38) and (3.6.39), $|| \ldots ||$ is the Euclidean norm.

The minimization problem implied by (3.6.38) is often

easily solved by equating to zero the partial derivatives
of the integral with respect to the components of $\underline{b}$; this
will yield a sufficient set of simultaneous equations in-
volving the components of $\underline{b}$. The solution of this set of
equations yields the specific optimal controller.

The minimization problem implied by (3.6.39) is more
difficult to solve and will be discussed in Chapter 6 of
this report.

Example 3.4:

Consider a second order plant described by the differ-
ential equations

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = -3x_2 - 2x_1 - 0.5x_1^3 + u$$

(3.6.40)

Let the plant be in an initial state

$$x_1(0) = C_1$$

$$x_2(0) = C_2$$

(3.6.41)

The performance index to be minimized is

$$I_1 = \int_0^T (x_1^2 + x_2^2 + u^2)\, dt$$

(3.6.42)

where T is the fixed terminal time.

For numerical results, let

$$C_1 = 2.0$$

$$C_2 = 2.0 \qquad (3.6.43)$$

$$T = 1.0$$

(i)   Open-loop optimal solution

The Lagrangian is

$$L = (x_1^2 + x_2^2 + u^2) + \lambda(x_2 - \dot{x}_1)$$

$$+ \mu(-3x_2 - 2x_1 - 0.5x_1^3 + u - \dot{x}_2)$$

The Euler-Lagrange equations are

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = -3x_2 - 2x_1 - 0.5x_1^3 - 0.5\mu$$

$$\dot{\lambda} = -2x_1 + 2\mu + 1.5\mu \, x_1^2 \qquad (3.6.44)$$

$$\dot{\mu} = -2x_2 - \lambda + 3\mu$$

$$u = -0.5 \, \mu$$

The last relation in (3.6.44) is an algebraic relation.

The boundary conditions on (3.6.44) are

$$x_1(0) = 2.0 \quad , \quad x_2(0) = 2.0$$

$$(3.6.45)$$

$$\lambda(1) = \mu(1) = 0$$

The set of differential equations (3.6.44) with boundary conditions (3.6.45) is solved by the method of quasilinearization. Let the solution be $u^*(t)$ and $x_1^* = \varphi(t)$.

The value of the performance index for the optimal solution is

$$I(u = u^*(t)) = 5.05815351 \qquad (3.6.46)$$

(ii) SOC ($u = b\, x_1$):

Let the only accessible state be $x = x_1$ and the desired SOC be of the form

$$u = bx \qquad (3.6.47)$$

Then the problem is to find the feedback coefficient b such that the solution of the differential equations

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = -3x_2 - 2x_1 - 0.5x_1^3 + bx_1$$

or

$$\ddot{x} + 3\dot{x} + 2x + 0.5x^3 - bx = 0 \qquad (3.6.48)$$

with the boundary conditions

$$x_1(0) \ = \ x(0) \ = \ c_1$$

$$x_2(0) \ = \ \dot{x}(0) \ = \ c_2$$

approximates the optimal trajectory $x^*(t)$.

Here, the value of b will be picked such that

$$\int_0^T (\ddot{\varphi} + 3\dot{\varphi} + 2\varphi + 0.5\varphi^3 - b\varphi)^2 \ dt \qquad\qquad (3.6.49)$$

is minimized with respect to b.

The minimization of the integral in (3.6.49) with respect to b results in the equation

$$b\left(\int_0^T \varphi^2 \ dt\right) \ = \ \left(\int_0^T (\ddot{\varphi} + 3\dot{\varphi} + 2\varphi + 0.5\varphi^3)\varphi \ dt\right) \qquad (3.6.50)$$

Consider the quantity in the parentheses in the integrand on the right-hand side of (3.6.50). The following relation is true:

$$\ddot{\varphi} + 3\dot{\varphi} + 2\varphi + 0.5\varphi^3 \ = \ u^*(t)$$

Thus

$$b \ = \ \frac{\displaystyle\int_0^T u^*(t) \ \varphi(t) \ dt}{\displaystyle\int_0^T \varphi^2(t) \ dt} \qquad\qquad (3.6.51)$$

The value of b for the initial conditions given in (3.6.43) is

$$b = 0.039054$$

and the value of the performance index is

$$I(u = bx_1) = 5.10033399$$

The specific optimal trajectories (x_1^S, x_2^S) are compared with the optimal trajectories (x_1^*, x_2^*) and are shown in Figure 3.4.

It is interesting to note that the specific optimal trajectory matches the optimal very closely. The percentage deviation in the index of performance with respect to the optimal solution is

$$\frac{I(u = u^*(t)) - I(u = bx)}{I(u = u^*(t))} \times 100$$

$$\approx 0.85\%$$

This indicates that in an engineering problem of this type an SOC of the form u = bx is sufficient. Moreover, this type of control is extremely simple to realize compared to the optimal control function. It is possible in some cases
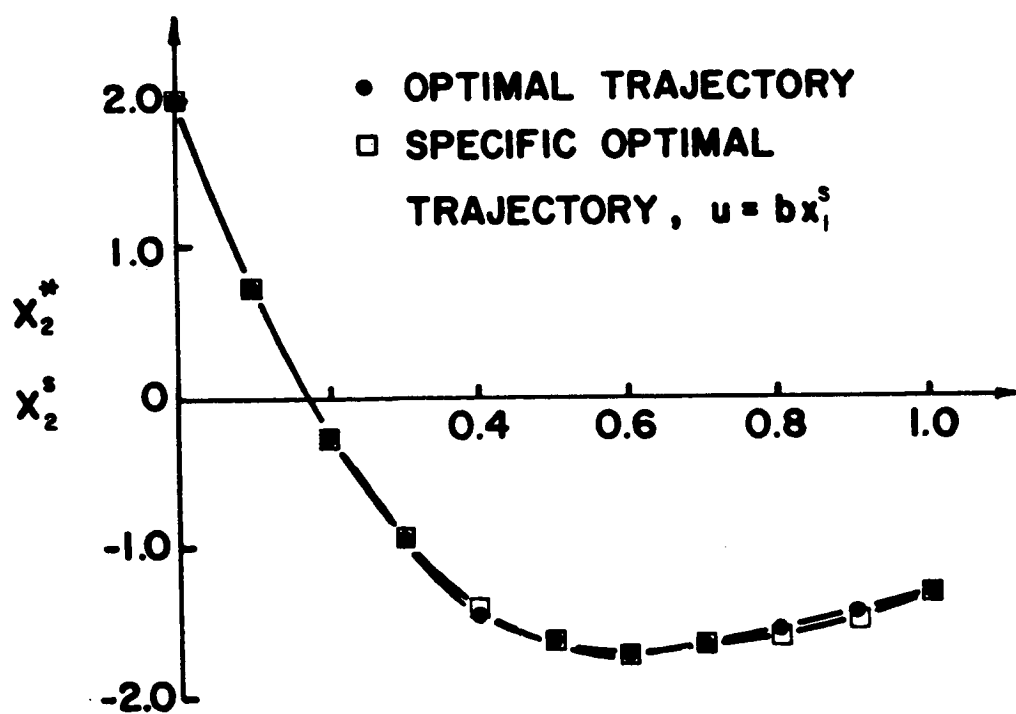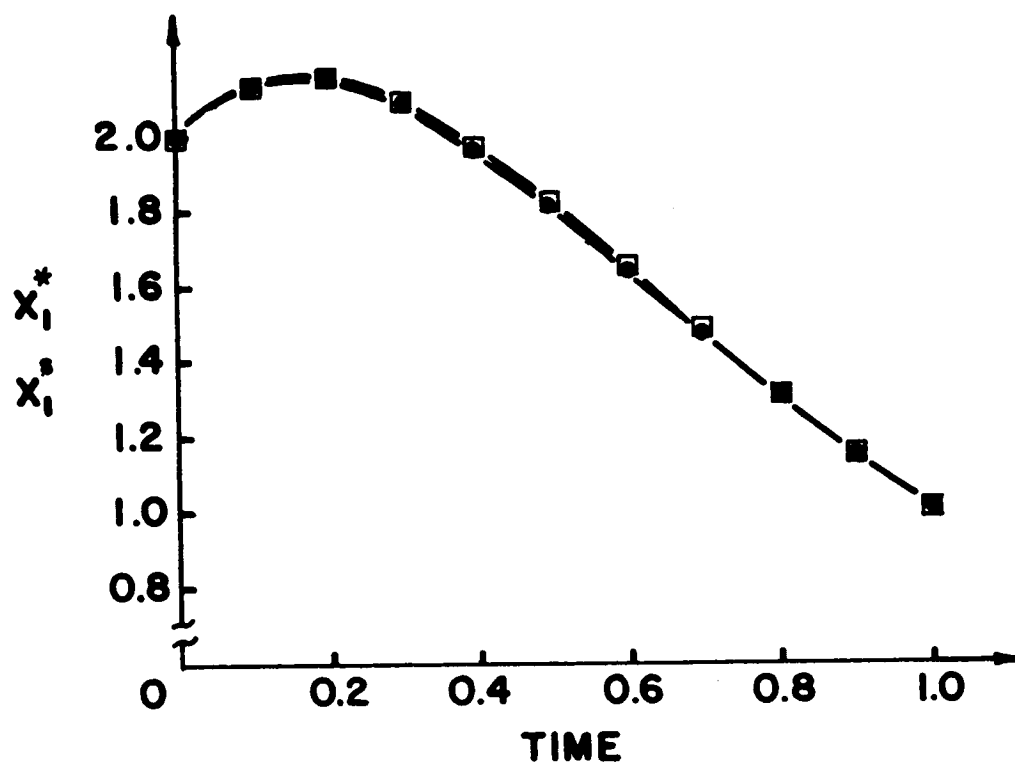
FIGURE 3.4 Specific Optimal Trajectory

Compared with the Optimal

to obtain better overall performance using an SOC as compared to the optimal because of the instrumentation problems in realizing the optimal control function.

(iii) SOC ($u = bx_1 + cx_1^3$):

Let the only accessible state be $x = x_1$, and let it be desired to determine how much improvement over the performance obtained above can be achieved by using a nonlinear controller, say of the form $u = bx_1 + cx_1^3$.

Here, the values of b and c are obtained such that

$$\int_0^T (\ddot{\varphi} + 3\dot{\varphi} + 2\varphi + 0.5\varphi^3 - b\varphi - c\varphi^3)^2 \, dt \qquad (3.6.52)$$

is minimized.

Notice, again

$$\ddot{\varphi} + 3\dot{\varphi} + 2\varphi + 0.5\varphi^3 = u^*(t)$$

Thus (3.6.52) reduces to

$$\underset{b, c}{\text{Min}} \int_0^T (u^*(t) - b\varphi - c\varphi^3)^2 \, dt \qquad (3.6.53)$$

The minimization results in the following set of linear equations:

$$\left(\int_0^T \varphi^2(t) \ dt \right) b + \left(\int_0^T \varphi^4(t) \ dt \right) c = \int_0^T u^*(t) \ \varphi(t) \ dt$$

<div align="right">(3.6.54)</div>

$$\left(\int_0^T \varphi^4(t) \ dt \right) b + \left(\int_0^T \varphi^6(t) \ dt \right) c = \int_0^T u^*(t) \ \varphi^3(t) \ dt$$

The solution of this set of equations yields

$$b = 0.287247$$

$$c = -0.068932$$

for the same initial conditions as in (ii), and the value
of the performance index is

$$I = 5.08265859$$

The percentage difference between this value of the
index of performance and the optimal value is

$$\frac{I(u = u^*(t) - I(u = bx_1 + cx_1^3)}{I(u = u^*(t))} \times 100$$

$$\approx 0.5\%$$

The improvement in the performance compared to the SOC
$u = bx_1$ is very small.

(iv) SOC $(u = bx_1 + cx_2)$:

Now suppose that the second state $x_2 (= \dot{x})$ is also accessible and it is desired to build a controller of the form

$$u = bx_1 + cx_2$$

This leads to the minimization of the integral

$$\underset{b,c}{\text{Min}} \int_0^T (u^*(t) - b\varphi(t) - c\dot{\varphi}(t))^2 \, dt \qquad (3.6.55)$$

The values of b and c are obtained as outlined above,

$$b = -0.046133$$

$$c = -0.178149$$

and the value of the index of performance is

$$I = 5.06101251$$

The percent deviation of this index of performance from the optimal value is

$$\frac{I(u = u^*(t)) - I(u = bx_1 + cx_2)}{I(u = u^*(t))} \times 100$$

$$\approx 0.06\%$$

This solution is very close to the optimal solution.

The specific optimal trajectories $(x_1{}^S, x_2{}^S)$ for the controllers $u = bx_1 + cx_1{}^3$ and $u = bx_1 + cx_2$ are very close to the optimal trajectories and are about the same as shown in figure 3.4.

The different controls, i.e. $u^*(t)$, $u = bx_1$, $u = bx_1 + cx_1{}^3$ and $u = bx_1 + cx_2$, are shown in figure 3.5.

It should be mentioned here that once the optimal solution to the problem has been obtained, different types of controllers which are easy to instrument can be obtained with little additional computation using differential approximation.

The methods of parameter optimization and transformation to TPBVP require an initial approximation for the unknown coefficient vector $\underline{b}$ and the convergence of these schemes depends on a good initial guess. One may try to combine the advantages of these techniques. For example, differential approximation demands very little machine time but gives only an approximate answer which may be used as an initial guess for the quasilinearization scheme. The quasilinearization method is an accurate technique with quadratic convergence properties, however it involves relatively long
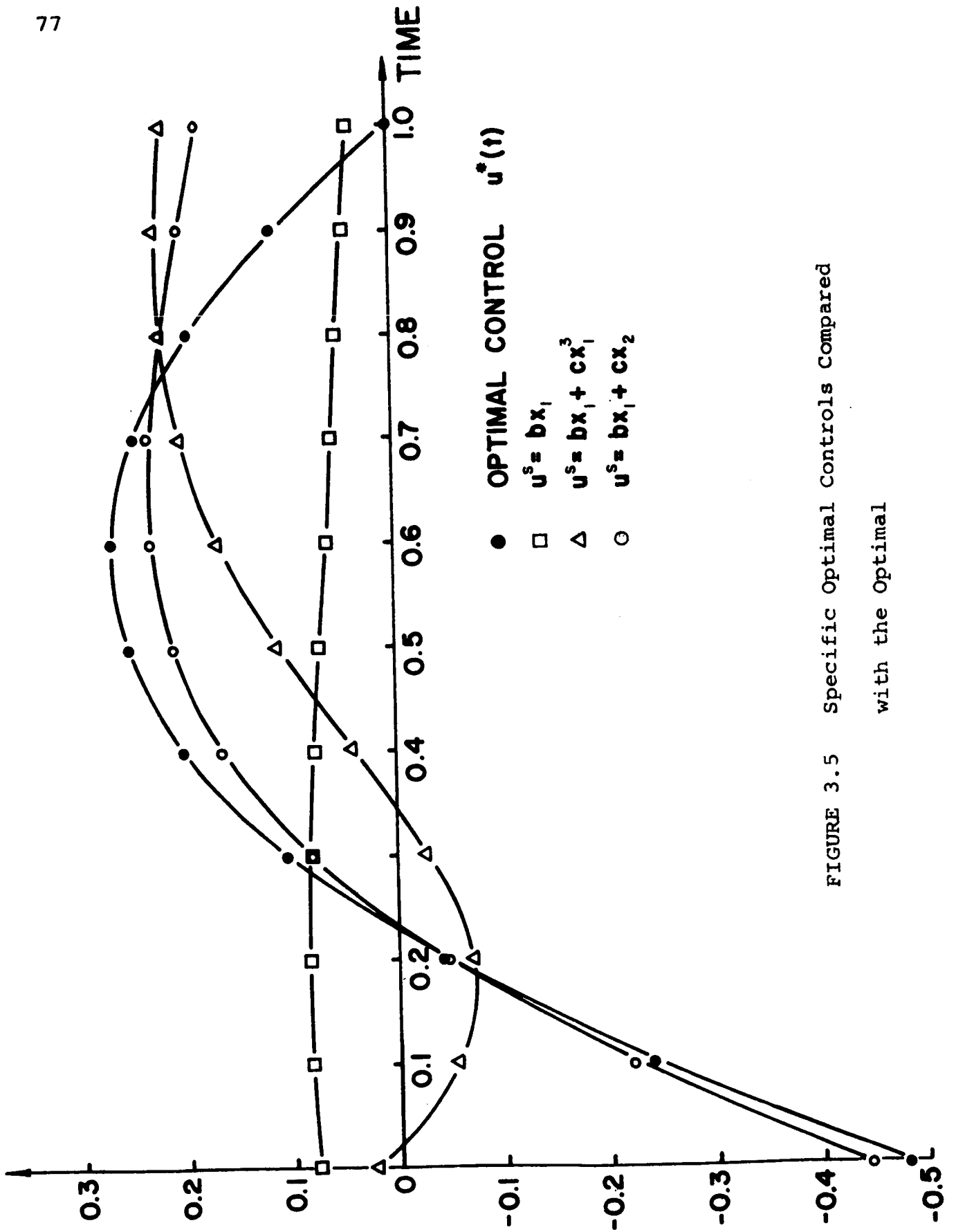
FIGURE 3.5   Specific Optimal Controls Compared
with the Optimal

computing times.

For example 3.4, the quasilinearization (Q.L.) and differential approximation (D.A.) are combined in the following manner:

(i)   Find open-loop optimal solution, i.e. solve the TPBVP (3.6.44) by Q.L.

(ii) For SOC $u = bx_1$, find b by D.A., then apply Q.L. for more accurate solution.

(III) Repeat part 2 for SOC $u = bx_1 + cx_1^3$ and $u = bx_1 + cx_2$.

For the initial conditions (3.6.43), the results are as follows:

(i)   Open-loop optimal solution.

   $I(u = u^*(t)) = 5.0581535$

(ii) SOC $(u = bx_1)$:

   D. A. scheme

   $b = 0.039054$ $I = 5.1003339$

   Q. L. scheme

   $b = 0.039605$ $I = 5.1003334$

(iii) SOC $(u = bx_1 + cx_1^3)$:

   D. A. scheme

   $b = 0.287247$ $c = -0.068932$

   $I = 5.0826586$

Q. L. scheme

b = 0.289593   c = -0.069367

I = 5.0826567

(iv) SOC ($u = bx_1 + cx_2$):

D. A. scheme

b = -0.046133   c = -0.178149

I = 5.0610125

Q. L. scheme

b = -0.046353   c = -0.178814

I = 5.0510125

This indicates that the D. A. solution in many cases should suffice unless a very accurate solution is desired.

A listing of the complete FORTRAN program, consisting of a main program and the derivative subroutines is given in appendix H.

3.7 Sensitivity Analysis

It was indicated in the examples that the feedback co-efficients in the specific optimal controller depend on the boundary conditions on the state variables and the duration of the process. For satisfactory implementation of the SOC controller, it is desirable that this dependence should be the

least possible. The study of this dependence is called the sensitivity analysis.

In different cases, different aspects of sensitivity analysis may be of interest. In example 3.2, the feedback coefficient b depends on C_1, C_2 and T. If the variations in these boundary conditions are small, it may be of interest to find the partial derivatives of b with respect to C_1, C_2 and T. Such an analysis can be made using the classical perturbation techniques [12]. In other cases, if the initial conditions can take values from a set (normally bounded) then it is desirable to study the variation of feedback coefficients over this set. For such studies, the invariant imbedding technique [13, 14] is often useful.

The variation of the feedback coefficients in the SOC problem also depends on the number of states available for manipulations. This is indicated by the following example.

Example 3.5

Consider the plant described by the equation

$$\ddot{x} + 3\dot{x} + 2x = u$$

or in state variable form

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = -3x_2 - 2x_1 + u \tag{3.7.1}$$

The performance index to be minimized is

$$I_1 = \int_0^\infty (x_1{}^2 + x_2{}^2 + u^2) \, dt \tag{3.7.2}$$

Let the plant be in the initial state

$$x_1(0) = C_1$$

$$x_2(0) = C_2 \tag{3.7.3}$$

The optimal trajectory for this problem is readily obtained by solving the canonic equations (equations (A.4.23) and (A.4.24) in appendix A). The optimal control function is

$$u^*(t) = -0.235(C_1 + C_2) \, e^{-2.236t} \tag{3.7.4}$$

and the corresponding optimal trajectory is

$$x_1{}^* = -0.81(C_1 + C_2) \, e^{-2.236t} + 0.81(2.236 \, C_1 + C_2) \, e^{-t} \tag{3.7.5}$$

$$x_2{}^* = 1.81(C_1 + C_2) \, e^{-2.236t} - 0.81(2.236 \, C_1 + C_2) \, e^{-t}$$

If both the states are available, and since the plant is linear with quadratic performance index, it is possible to write the control function (3.7.4) as

$$u^* = -0.235x_1 - 0.235x_2 \qquad (3.7.6)$$

In this case the feedback coefficients are independent of the initial conditions C_1 and C_2.

However, if only one state is available and the desired SOC is of the form $u = bx_1$ then by the differential approximation method, find b as

$$\underset{b}{\text{Min}} \int_0^\infty [\ddot{x} + 3\dot{x} + (2-b)x]^2 \, dt \qquad (3.7.7)$$

This minimization results in

$$b = -\frac{(0.1n^2 + 0.14n + 0.024)}{(0.877n^2 + 0.4396n + 0.0693)} \qquad (3.7.8)$$

where $n = C_1/C_2$. Figure 3.6 portrays the dependence of b on the ratio n. The value of b is fairly constant except in the range where C_1/C_2 is small. This type of behaviour one would expect since the variable fed-back has relatively small magnitude compared to the second variable.
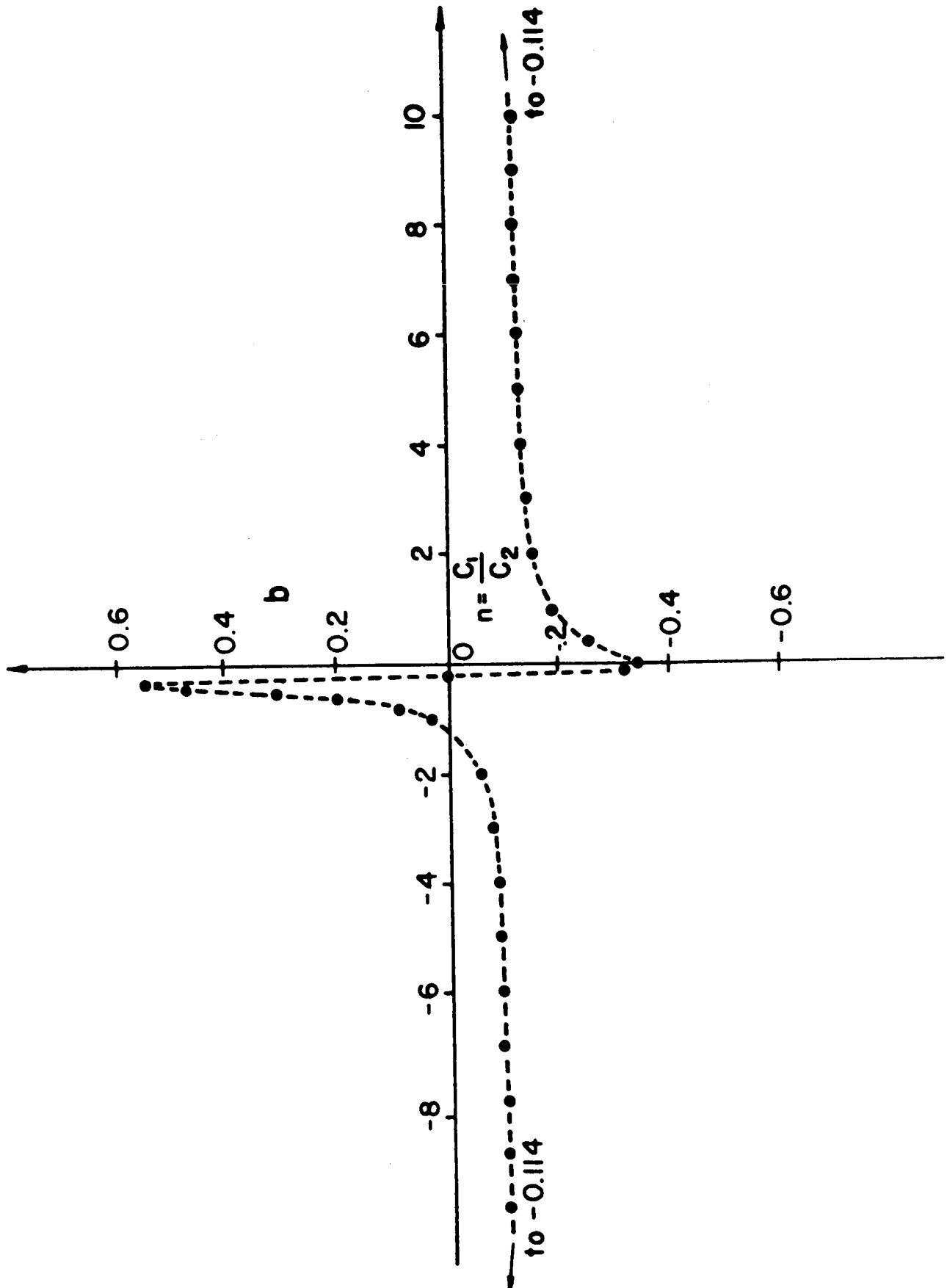
Figure 3.6 Dependence of the Feedback Coefficient b on the ratio of Initial Conditions n.

Example 3.6

Consider the same plant as in example 3.5, i.e.

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = -3x_2 - 2x_1 + u \qquad (3.7.9)$$

with the initial conditions

$$x_1(0) = C_1$$

$$x_2(0) = C_2$$

Let the performance index to be minimized be

$$I_1 = \int_0^1 (x_1^2 + x_2^2 + u^2) \, dt \qquad (3.7.10)$$

Let x_1 be the only available state and let the initial conditions be in the range -2 to 2.

If the initial condition on x_2 is unknown and it is assumed that C_2 can take values from -2 to 2 with equal likelihood, then it may be desirable to study the variation of the feedback coefficients for different values of C_1, taking $C_2 = 0$.

For the SOC $u = bx_1$ the feedback coefficient b turns out to be a constant and equal to 0.024502 for $|C_1| \leq 2$ and $C_2 = 0$. If an improvement in the index of performance is

desired, consider an SOC of the form $u = bx_1 + cx_1^3$. In this case b is constant again and is equal to 0.210114 but c depends on the value of C_1 as shown in figure 3.7. This type of behavior for c results because when C_1 is small (less than 1), x_1^3 is very small and a large value of feedback gain c is necessary to get any contribution from the cx_1^3 term. Since the improvement in the index of performance is not significant, it is obvious that the $u = bx_1 + cx_1^3$ controller is not satisfactory compared to $u = bx_1$.

3.8 Conclusions

The space age optimum control problems are so complex that it becomes necessary to incorporate limitations on the measurements and restrictions on the controller for reliability and physical realizability in the optimization problem. The specific optimal control formulation is one way of attacking such problems. It was shown that deterministic disturbances can be taken care of in designing the controller. To make the SOC approach more meaningful, a study of systems with unknown disturbances is highly desirable. If some characteristics of the disturbances can be obtained, it may be desirable to design a specific optimal controller for the worst case disturbances.
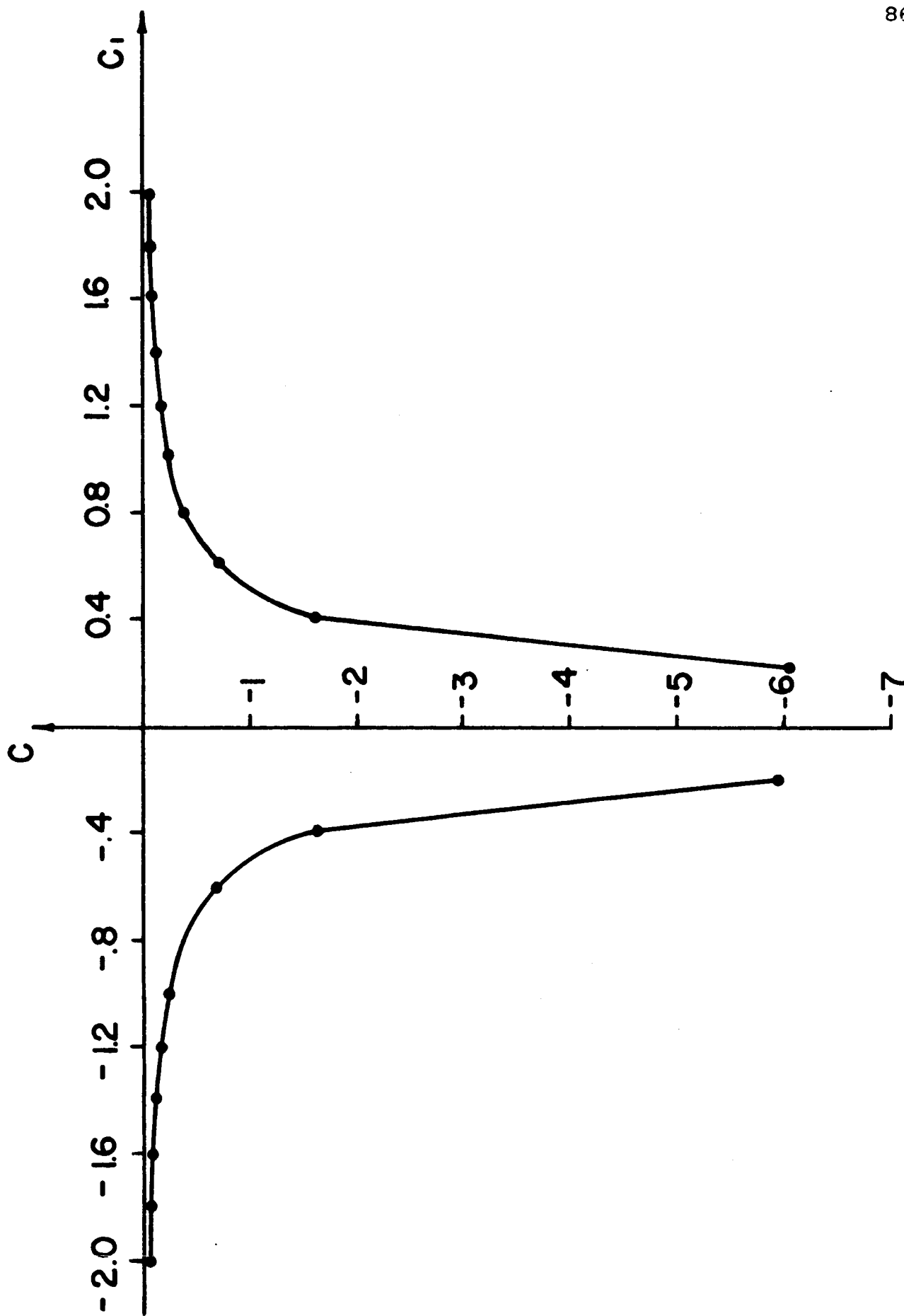
FIGURE 3.7   Dependence of C on the initial state C_1.

$C_2 = 0$ and $u = bx_1 + cx_1^3$

It was pointed out that the sensitivity problem for SOC is of importance. Some numerical approaches to this problem were presented; however, it is highly desirable to develop analytical tools to study such problems.

CHAPTER 4

BOUNDED CONTROL PROBLEMS

4.1 Summary

In this chapter the optimum control problem involving
constraints on the control variables is discussed. A brief
summary of existing methods to solve bounded control prob-
lems is given. The necessity of producing the optimum con-
trol function in these problems is explained, and a compu-
tational algorithm using "approximation in policy space" for
obtaining the control function is proposed. This algorithm
is applied to a number of examples including the space vehi-
cle attitude control problem. Future research and investi-
gations are outlined.

4.2 Introduction

A number of examples can be given in order to illustrate
a typical bounded control problem. Consider, for instance,
the problem of controlling the attitude of a space vehicle.
The space vehicle dynamics can be represented by a set of
six ordinary differential equations of the form

$$\dot{\underline{\omega}} = \underline{f} \; (\underline{\omega}, \; \underline{\alpha}, \; \underline{u}) \tag{4.2.1}$$

$$\dot{\underline{\alpha}} = \underline{g} \; (\underline{\omega}, \; \underline{\alpha})$$

where $\underline{f}$ and $\underline{g}$ are vector functions of the states $\underline{\omega}$ and $\underline{\alpha}$. $\underline{\omega}$ is the angular velocity vector, $\underline{\alpha}$ the Euler angle vector and $\underline{u}$ the control torque vector.

The object is to synthesize a control law which will transfer the space vehicle from any initial state to a desired final state in a way such that a performance measure is extremized. Also in attitude control problems an additional restriction is that the magnitude of the components of the control input $\underline{u}$ be bounded due to the physical restriction that the jets can deliver only a certain level of torque. This added constraint on the control usually makes it more difficult to obtain the solution. In order to illustrate the various methods which exist to solve the bounded control problem it is best to formulate the problem in general terms.

4.3 Methods of Solution [5,17,21,24,15,App. A]

Let the dynamical system under consideration be represented by the following first order vector differential equation of the form

$$\dot{\underline{x}} = \underline{f}(t, \underline{x}, \underline{u}) \qquad \underline{x}(0) = \underline{x}_o \qquad (4.3.1)$$

where $\underline{x}$ is the n-dimensional vector state

and $\underline{u}$ is the m-dimensional vector control

Constraints on the control are usually of the form

$$K_{1j} \leq u_j \leq K_{2j} \qquad j = 1, 2, \cdots m \qquad (4.3.2)$$

Consider the performance measure to be of the type

$$\int_o^T g(t, \underline{x}, \underline{u}) \, dt \qquad (4.3.3)$$

where $g(t, \underline{x}, \underline{u})$ is a non-negative scalar function of its arguments. T, the duration of the process, will be considered to be fixed. The object is to find $\underline{u}$ such that (4.3.3) is minimized subject to (4.3.1) and (4.3.2). The following methods can be utilized to solve the above problem:

    (i)   Dynamic Programming

    (ii)   Hamilton-Jacobi Equation

    (iii)   Calculus of Variations

    (iv)   Pontryagin's Maximum Principle

In order to illustrate the methods (i) to (iv) above consider the simple scalar example given below. Let the system be governed by the equation

$$\dot{x} = ax + u \qquad (4.3.4)$$

with $\qquad x(0) = x_o$

The constraint on u is

$$|u| \leq 1 \qquad (4.3.5)$$

The performance measure to be minimized is

$$\int_0^T (x^2 + u^2) \, dt \qquad (4.3.6)$$

(i) <u>Dynamic Programming</u> [5,15,18,26]

To solve the problem as defined by (4.3.4),(4.3.5) and (4.3.6), consider the class of processes of the variational problem of arbitrary initial state C and initial time τ. In other words one wishes to minimize the functional

$$I(u) = \int_\tau^T (x^2 + u^2) \, dt \qquad (4.3.7)$$

with $x(\tau) = C$

The functional $I(u)$ subject to (4.3.4) and (4.3.5) can be thought of as the cost of the process which clearly depends on C and τ and $u(t)$, $t \in [\tau, T]$. The minimum of $I(u)$ over all allowable u depends only on C and τ. Hence define the value function

$$J(C, \tau) \overset{\Delta}{=} \underset{|u| \leq 1}{\text{Min}} \ [I(u)] \qquad (4.3.8)$$

Applying Bellman's principle of optimality, one obtains

$$J(C, \tau) = \underset{|u| \leq 1}{\text{Min}} \ \left[(C^2 + u^2)\Delta + J(C + \Delta C, \ \tau + \dot{c}) \right] \qquad (4.3.9)$$

The terminal condition becomes $J(C, T) = 0$ from (4.3.8).
The recurrence relation (4.3.9) can be solved by search techniques and one can in principle obtain the control law, i.e.
$u = u(t, x)$. The main difficulty in this method is the
'curse of dimensionality' [18] which for higher dimensional
problems makes the solution impossible.

(ii)   <u>Hamilton-Jacobi Equation</u> [1]

The Hamiltonian for the problem is

$$H(x, \lambda, u) \overset{\Delta}{=} \lambda u + \lambda a x + (x^2 + u^2) \qquad (4.3.10)$$

where λ is the multiplier. The value function $J(t, x)$ is
defined exactly as in equation (4.3.8) to be

$$J(x, t) = \underset{|u| \leq 1}{\text{Min}} \ \int_t^T (x^2 + u^2) \ dt \qquad (4.3.11)$$

93

A slight change of notation is evident when equations (4.3.8) and (4.3.11) are compared. The value function in eq. (4.3.11) is associated with the cost of a process starting in state x at time t.

The Hamilton-Jacobi Equation for the problem is

$$\frac{\partial J}{\partial t} + H^* \ (x, \ \lambda) \qquad\qquad (4.3.12)$$

where H* is the Hamiltonian which has been minimized with respect to u. The multiplier λ can be written in terms of J as $\lambda = \partial J / \partial x$ (generally $\lambda = \nabla_x J$)

The u which minimizes the Hamiltonian is then,

$$u^* = \begin{cases} -\frac{1}{2} \frac{\partial J}{\partial x} & \left| \frac{1}{2} \frac{\partial J}{\partial x} \right| \leq 1 \\[2ex] \text{sgn}\left[-\frac{1}{2} \frac{\partial J}{\partial x} \right] & \left| \frac{1}{2} \frac{\partial J}{\partial x} \right| \geq 1 \end{cases} \qquad (4.3.13)$$

Hense, $H^*(x, \frac{\partial J}{\partial x}) = \varphi(\frac{\partial J}{\partial x}, \ x) + x^2$ (4.3.14)

where $\varphi(\frac{\partial J}{\partial x}, \ x) = \frac{\partial J}{\partial x} u^* - u^{*2} + \frac{\partial J}{\partial x} ax$ (4.3.15)

From (4.3.14), one can rewrite (4.3.12) as

$$\frac{\partial J}{\partial t} + \varphi(\frac{\partial J}{\partial x}, x) + x^2 = 0 \qquad (4.3.16)$$

The boundary condition for (4.3.16) is

$$J(T, x) = 0 \qquad (4.3.17)$$

from the definition (4.3.11). The solution of the partial differential equation (4.3.16) subject to the condition (4.3.17) will provide the control law, i.e. u = u(t, x). Equations (4.3.16) and (4.3.17) are very difficult to solve in general and analytical solution is almost impossible.

(iii) <u>Calculus of Variations</u> [21,24]

To attempt a solution via the calculus of variations one transforms the inequality constraints of u to an equality constraint. This is done in the following fashion. Define an auxiliary variable z such that

$$z^2 = 1 - u^2 \qquad (4.3.18)$$

The Lagrangian of the variational problem is

$$L = x^2 + u^2 + \lambda_1(ax + u - \dot{x}) + \lambda_2(1-u^2-z^2) \qquad (4.3.19)$$

where λ_1 and λ_2 are Lagrange multipliers.

The associated Euler equations are

$$\dot{x} = ax + u$$

$$\dot{\lambda_1} = -\lambda_1 - 2x$$

$$0 = \lambda_1 + 2(1-\lambda_2)u \qquad (4.3.20)$$

$$0 = 1 - u^2 - z^2$$

$$0 = \lambda_2 z$$

The boundary conditions are $x(0) = C$, $\lambda_1(T) = 0$.

The analytical solution of (4.3.20) along with the boundary conditions is quite impossible in general and one must resort to numerical techniques in order to obtain a solution.

(iv) <u>Pontryagin's Maximum Principle</u> [21,24]

The Hamiltonian for the problem is

$$H \overset{\Delta}{=} \lambda ax + \lambda u + x^2 + u^2 \qquad (4.3.21)$$

The associated canonic equations are

$$\dot{x} = ax + u^*$$
$$\dot{\lambda} = -a\lambda - 2x \qquad (4.3.22)$$

where u^* is obtained by minimizing the Hamiltonian with respect to u over the interval $0 \leq t \leq T$. Then u^* is

$$u^* = \begin{cases} -\dfrac{\lambda}{2} & \left| \dfrac{\lambda}{2} \right| \leq 1 \\ \\ \mathrm{sgn}\left(\dfrac{-\lambda}{2}\right) & \left| \dfrac{\lambda}{2} \right| \geq 1 \end{cases} \qquad (4.3.23)$$

One of the boundary conditions for (4.3.22) is $x(0) = C$; the other one is obtained from the transversality condition for the variational problem and is $\lambda(T) = 0$. The solution of the two point boundary value problem (4.3.22) provides the control function $u = u^*(t)$. In many cases the elimination of u^* from the canonic equations (4.3.22) is difficult since u^* may not be explicitly determined in terms of λ and x using the maximum principle. Even when u^* can be eliminated, one faces the formidable task of solving a two point boundary value problem.

From the foregoing explanation of existing methods, one finds that it is indeed difficult to produce the optimum u, either as a control function or control law even for the simple scalar example. In order to overcome these difficulties, it is apparent that one may have to resort to specific controllers.

4.4 Specific Control [7,8,27]

In proposing a specific control for any problem, the choice is dependent upon the ease of instrumentation, cost of controller and the number of accessible states. The final design will rely heavily on the type of performance obtained,

97

for example the difference in performance index value from the optimal. In such a design one has to realize that compromises have to be made when one has to decide between two conflicting interests. It should be noted that for the bounded control case, the specific controller should also satisfy the constraints on u.

Consider the same scalar example as in section 4.3. A specific controller that one may propose can take the form

$$u = \text{sat}(ax) \qquad (4.4.1)$$

where

$$\text{sat}(ax) = \begin{cases} ax & |ax| \leq 1 \\ \text{sgn}(ax) & |ax| \geq 1 \end{cases} \qquad (4.4.2)$$

It is seen that (4.4.1) satisfies the constraints on u, and one wishes to find the unknown parameter "a" so as to minimize the performance measure given by (4.3.6). One of the methods by which the optimum "a" can be found is by parameter optimization. A number of techniques are available for doing this [5,7,8].

A different approach could be taken in proposing specific controllers. One can change the "hard constraint" on the control to a somewhat equivalent "soft constraint."

This is done as follows. Remove the inequality constraint (hard constraints) on u. Impose a heavy penalty in the performance measure for deviations from the hard constraint. For example the performance measure given by (4.3.6) is changed to

$$\int_0^T (x^2 + u^2 + \alpha u^{2N}) \, dt \qquad (4.4.3)$$

where α is a large positive constant and N a positive integer greater than unity.

Now along with (4.4.3) one can propose an unconstrained specific control of the form

$$u = bx \qquad (4.4.4)$$

Due to the weighting on u which imposes a heavy penalty whenever the magnitude of u exceeds unity in (4.4.3) one can intuitively expect that the optimum u obtained by minimizing (4.4.3) to be almost constrained within the bounds. The problem now has been reduced to the SOC problem of Chapter 3 and the solution can be effected by any one of the methods given therein. The choice of values for α and N will depend to a large extent on the loss of performance and the magnitude of violation of the constraints on u which can be tolerated.

Before any synthesis procedures for these specific controllers are attempted one clearly sees the necessity of obtaining the optimal solution so that one would have a "yardstick" for comparison in design. As seen from section 4.3, the optimum solution is quite difficult to obtain analytically. Hence, one has to look for effective computational algorithms in order to obtain the optimal solution. One such algorithm will be discussed in the next section.

4.5 Approximation in Policy Space [15,18,26,28]

Consider the general set of system equations

$$\dot{\underline{x}} = \underline{f}(t, \underline{x}, \underline{u}) \qquad (4.5.1)$$

$$\text{with } \underline{x}(0) = \underline{x}_o$$

Let the constraint on $\underline{u}$ be of the form

$$|u_i| \leq U \qquad (4.5.3)$$

The performance measure to be minimized is

$$\int_o^T g(t, \underline{x}, \underline{u}) \, dt \qquad (4.5.3)$$

and T is fixed.

To solve this variational problem one writes the Hamiltonian as

$$H(t, \underline{x}, \underline{u}, \underline{\lambda}) = \langle \underline{\lambda}, \underline{f} \rangle + g(t, \underline{x}, \underline{u}) \qquad (4.5.4)$$

where λ is the n-dimensional multiplier vector. The associated canonic equation are

$$\dot{\underline{x}} = \underline{f}(t, \underline{x}, \underline{u}^*)$$

$$\dot{\lambda} = - \underline{H}_x^* \qquad (4.5.5)$$

where $\underline{u}^*$ is the $\underline{u}$ which minimized the Hamiltonian at each instant of time over the interval $0 \le t \le T$ and H* is $H(t, \underline{x}, \underline{u}^*, \underline{\lambda})$.

In (4.5.5)

$$\underline{H}_x^* = \left(\frac{\partial H^*}{\partial x_1} , \frac{\partial H^*}{\partial x_2} \cdots \frac{\partial H^*}{\partial x_n} \right) \qquad (4.5.6)$$

The boundary conditions are

$$\underline{x}(0) = \underline{x}_0 , \quad \underline{\lambda}(T) = \underline{0} \qquad (4.5.7)$$

from (4.5.1) and the transversality condition of the variational problem.

The algorithm proposed utilizes the following procedure.

Choose an initial guess on $\underline{u}$ denoted by $\underline{u}^o$ and solve the

TPBVP given by (4.5.5) and (4.5.7), obtaining initial solu-

tions for $\underline{x}^o$ and $\underline{\lambda}^o$, respectively. Now utilize the Maximum

Principle and minimize (4.5.4) with respect to $\underline{u}$ to obtain

the first approximation $\underline{u}^1$. This approximation is utilized

to produce the trajectories $\underline{x}^1$ and $\underline{\lambda}^1$ using (4.5.5) and

(4.5.7).

This process is repeated until convergence results, and

$\underline{u}^*(t)$ is produced along with the optimal trajectory $\underline{x}^*$. The

minimization procedure is computationally very simple as the

search procedure has only to scan over a bounded set of $\underline{u}$

values due to constraints on $\underline{u}$ given by (4.5.2). As the

procedure involves an initial guess on $\underline{u}$, the "policy", and

successive approximations on it, it is called "approximation

in policy space."

In order to illustrate this method consider the follow-

ing simple example.

Example 4.1

Consider the second order non-linear plant with a time

varying input described by

$$\dot{x}_1 = x_2$$

(4.5.8)

$$\dot{x}_2 = -3x_2 - 2x_1 - 0.5x_1^3 + 0.1 \sin 10t + u$$

Let the initial conditions be

$$x_1(0) = C_1$$

$$x_2(0) = C_2$$

(4.5.9)

The performance measure to be minimized is

$$\int_0^5 (x_1{}^2 + x_2{}^2 + u^2)\ dt$$

(4.5.10)

The constraint on u is

$$|u| \leq 1$$

(4.5.12)

The Hamiltonian for the variational problem is

$$H = (x_1{}^2 + x_2{}^2 + u^2) + \lambda_1 x_2 + \lambda_2(-3x_2 - 2x_1 - 0.5x_1{}^3$$

$$+\ 0.1 \sin 10t + u)$$

(4.5.12)

The canonic equations are

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = -3x_2 - 2x_1 - 0.5x_1{}^3 + 0.1 \sin 10t + u^*(t)$$

$$\dot{\lambda}_1 = -2x_1 + 2\lambda_2 + 1.5\lambda_2 x_1{}^2$$

$$\dot{\lambda}_2 = -2x_2 - \lambda_1 + 3\lambda_2$$

(4.5.14)

The boundary conditions are

$$x_1(0) = C_1 \qquad\qquad \lambda_1(5) = 0$$

$$\tag{4.5.15}$$

$$x_2(0) = C_2 \qquad\qquad \lambda_2(5) = 0$$

The Hamiltonian being a minimum for u* implies that

$$u^* = \begin{cases} -\dfrac{1}{2}\lambda_2 & \text{for} & \left|\dfrac{\lambda_2}{2}\right| \leq 1 \\[2em] +1 & \text{for} & \dfrac{\lambda_2}{2} < -1 \\[2em] -1 & \text{for} & \dfrac{\lambda_2}{2} > 1 \end{cases} \tag{4.5.16}$$

In many cases it might not be even possible to represent u* as a function of multipliers λ_1 and λ_2.

The solution to this problem is obtained in the following manner.

If one has the current approximation to u*(t) denoted by $u_n^*(t)$, the corresponding functions $x_{1n}(t)$, $x_{2n}(t)$, $\lambda_{1n}(t)$, $\lambda_{2n}(t)$ are found as explained previously, and the next approximation $u_{n+1}^*(t)$ is found by a search method or from an analytical expression.

The basic plan involves using quasilinearization for solving the two point boundary value problem.

$$\dot{x}_{1,n} = x_{2,n}$$

$$\dot{x}_{2,n} = -3x_{2,n} - 2x_{1,n} - 1.5x_{1,n-1}^2 x_{1,n}$$

$$- x_{1,n-1}^3 + u_n^*(t) + 0.1 \sin 10t$$

$$\dot{\lambda}_{1,n} = -2x_{1,n} + 2\lambda_{2,n} + 1.5x_{1,n-1}^2 \lambda_{1,n-1}$$

$$+ 3\lambda_{2,n-1} x_{1,n} - 3\lambda_{2,n-1} x_{1,n-1}$$

$$\dot{\lambda}_{2,n} = -2x_{2,n} - \lambda_{1,n} + 3\lambda_{2,n} \qquad (4.5.17)$$

with the boundary conditions

$$x_{1,n}(0) = c_1$$

$$x_{2,n}(0) = c_2$$

$$\lambda_{1,n}(5) = 0$$

$$\lambda_{2,n}(t) = 0$$

for the determination of x_{1n}, x_{2n}, λ_{1n}, λ_{2n}. In equation (4.5.17) u_n^* is the current approximation and is to be thought of as being a fixed function of time while solving the quasilinear equations. Notice that an initial approximation on $u_o^*(t)$ is necessary to start the iterative calculations.

The (n+1)-st approximation to u*(t) could be found either by a search of

$$\text{Min}_{u} \left[u^2 + \lambda_{2n} u \right] , \quad u = K_1 \Delta_1 \tag{4.5.18}$$

or by the analytical expression (4.5.16).

For the numerical solution let the initial conditions be

$$x_1(0) = 5.0 \qquad x_2(0) = 8.0$$

The numerical procedure explained above is now applied, and convergence to the solution of the original problem occurs in 3 iterations. The system trajectory $x^*(t)$ and the control $u^*(t)$ are given in Figures 4.1 and 4.2 respectively.

4.6 Results and Comparisons [App. I]

In this section, a few selected examples are considered and the comparative results are illustrated.

Example 4.2

Consider the simple linear system of the form

$$\begin{aligned}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= -2x_1 - 3x_2 + u
\end{aligned} \tag{4.6.1}$$

The performance measure to be minimized is

$$\int_0^1 (x_1^2 + x_2^2 + u^2) \, dt \tag{4.6.2}$$

PLANT EQUATIONS

$$\dot{x}_1 = x_2$$
$$\dot{x}_2 = -2x_1 - 3x_2 - 0.5x_1^3$$
$$+ 0.1 \sin 10t + u$$

$$P.I. = \int_0^5 (x_1^2 + x_2^2 + u^2) \, dt$$
$$= 54.033$$

I.C.

x(t)
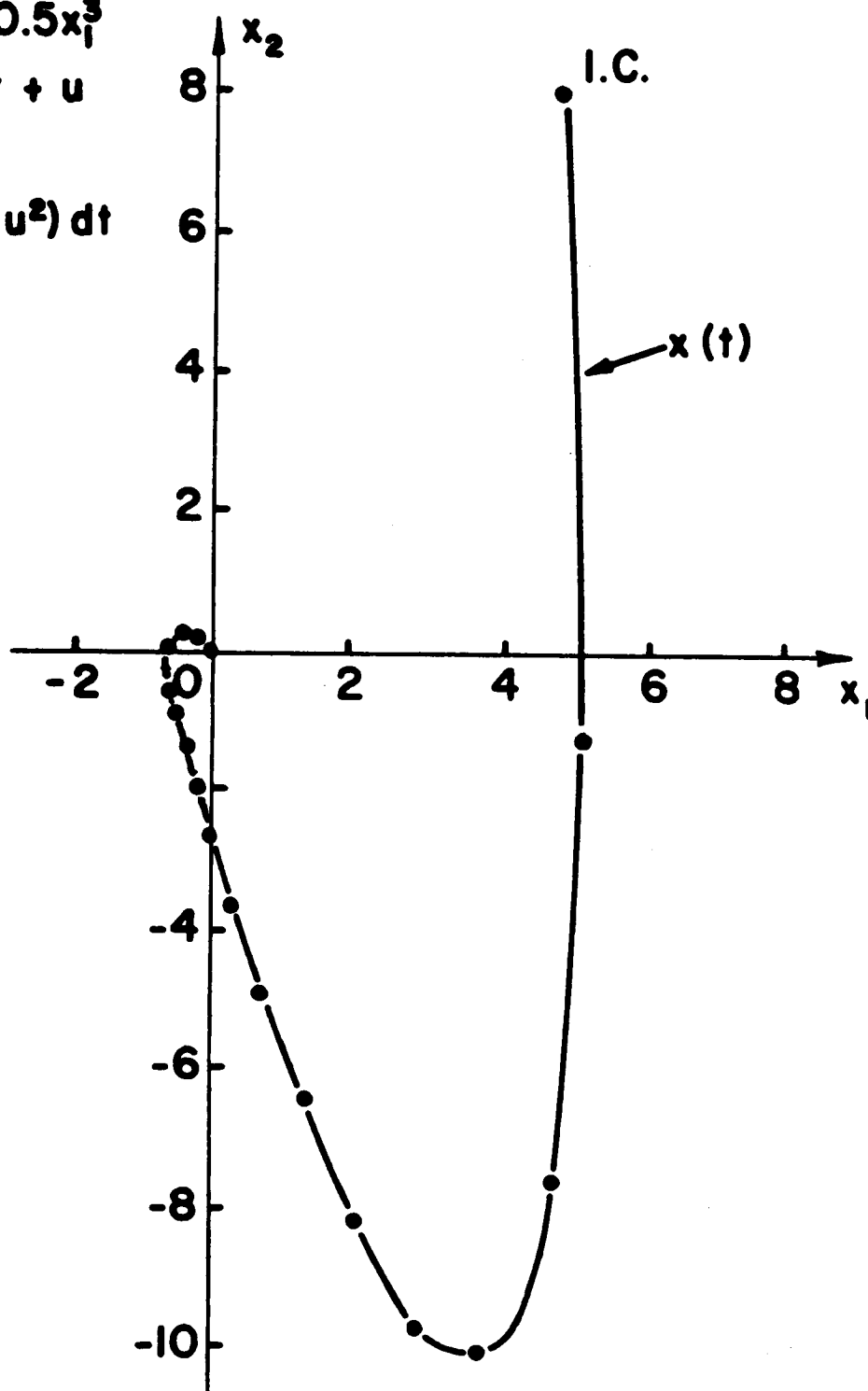
FIGURE 4.1
Optimal trajectory $\underline{x}^*$ for Example 4.1
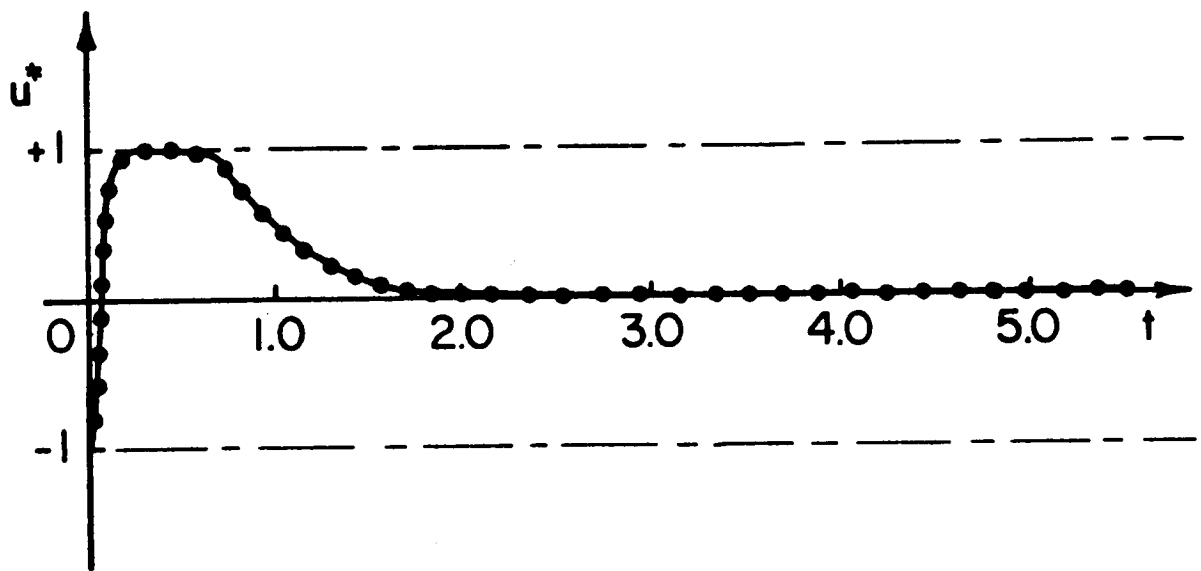
FIGURE 4.2

Optimal Control u*(t) for Example 4.1

with the constraint on u being

$$|u| \leq 1 \qquad\qquad (4.6.3)$$

In figures 4.3 and 4.4, the performance of this system for two different specific controllers is given for the initial conditions $x_1(0) = 5.0$, $x_2(0) = 8.0$. Also the optimal trajectory $\underline{x}^*$ and control u^* are plotted for comparison.

The two types of specific controllers are (i) $u = sat(ax_1 + bx_2)$ [the sat function is defined in (4.4.2)] (ii) $u = bx_1$ along with the soft constraint in the performance measure of the type

$$\int_0^1 (x_1{}^2 + x_2{}^2 + u^2 + u^4) \, dt$$

The values of the performance indices corresponding to the two types of controllers are as follows.

(i) $u = sat(ax_1 + bx_2)$

P.I. = 42.3847

a   = -0.0418

b   = -0.2007

(ii) $u = bx_1$ along with the soft constraint term in the integrand of the redefined performance index

PLANT EQUATIONS

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = -3x_2 - 2x_1 + u$$

$$P.I. = \int_0^1 (x_1^2 + x_2^2 + u^2)\, dt$$
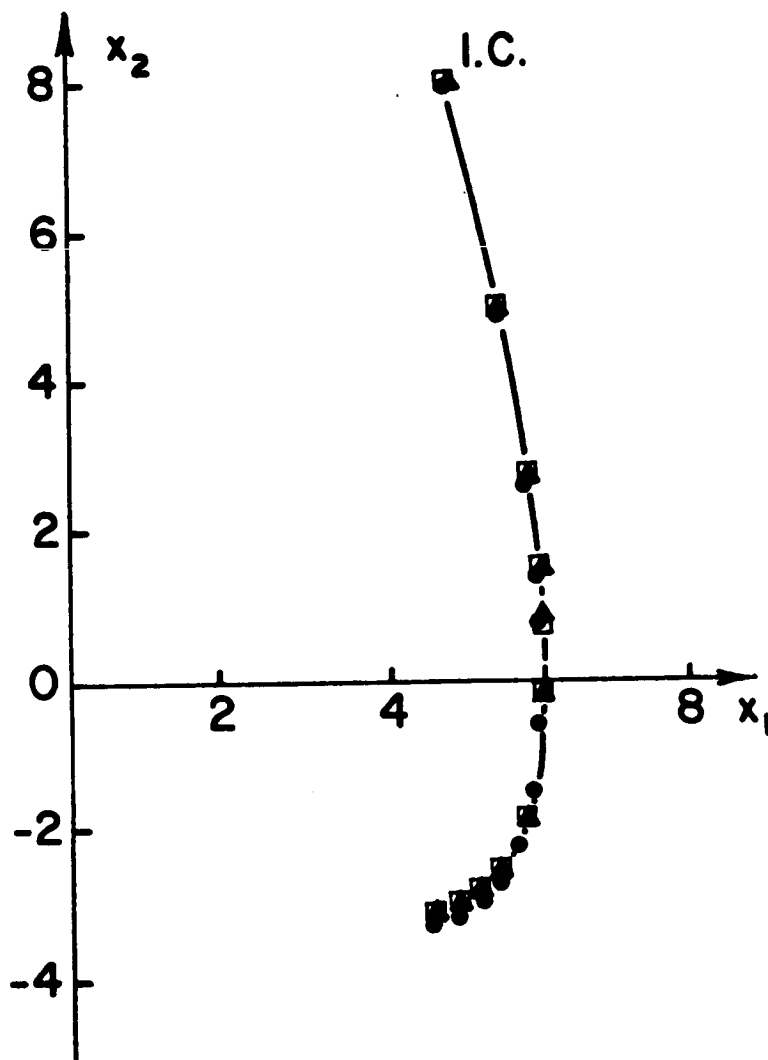
FIGURE 4.3

Comparison of trajectories for Example 4.2

$$\bullet \quad u^*(t), \ |u| \le 1$$

$$\blacktriangle \quad u_{s_1} = Sat \ (ax_1 + bx_2)$$
$$a = -0.0418$$
$$b = -0.2007$$

$$\blacksquare \quad u_{s_2} = bx_1$$
$$b = -0.0403$$
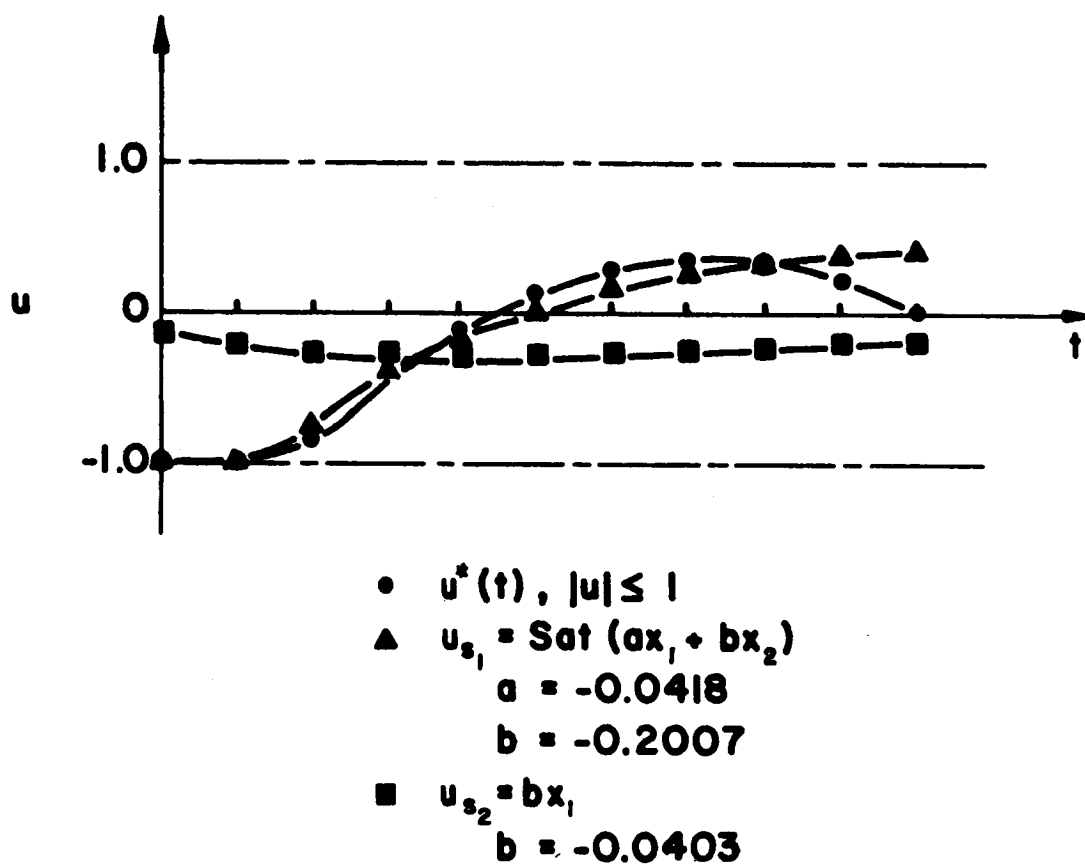
FIGURE 4.4

of the form

$$\int_0^1 (x_1{}^2 + x_2{}^2 + u^2 + u^4) \, dt$$

P.I. = 42.7768

b = -0.0403

It is to be pointed out that P.I.'s for (i) and (ii) are calculated using (4.6.2) even though the solution for (ii) uses the idea of soft constraint as given above, so that comparisons are compatible. The optimum performance index for this problem is

P.I.* = 42.3790

It is quite apparent that specific controller (i) is superior to (ii) as far as performance in concerned, but as pointed out earlier, the final choice of controllers can be made only after weighing the various factors, such as cost, simplicity of instrumentation and performance. It is also intuitively evident that (i) should be better than (ii) since information about both the states is utilized by the specific controller in (i). The programs utilized to produce the different _____ are listed in Appendix I.

Example 4.3

Consider now a non-linear system of the form

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = -2x_1 - 3x_2 - 0.5x_1{}^3 + u$$

(4.6.4)

The initial conditions are

$$x_1(0) = 5.0 \qquad x_2(0) = 8.0$$

The performance index to be minimized is

$$\int_0^1 (x_1{}^2 + x_2{}^2 + u^2) \, dt$$

(4.6.5)

and u is constrained by

$$|u| \leq 1$$

(4.6.6)

The same controllers used in Example 4.2 are considered and the results are given in Figures 4.5 and 4.6.

The performance indices and the parameters of the controllers are

(i) $u = sat(ax_1 + bx_2)$

P.I. = 52.8778

a   = -0.0212

b   = -0.1431

113

PLANT EQUATIONS

$\overset{\circ}{x}_1 = x_2$

$\overset{\circ}{x}_2 = -3x_2 - 2x_1 - 0.5x_1^3 + u$

$P.I. = \int_0^1 (x_1^2 + x_2^2 + u^2)\,dt$
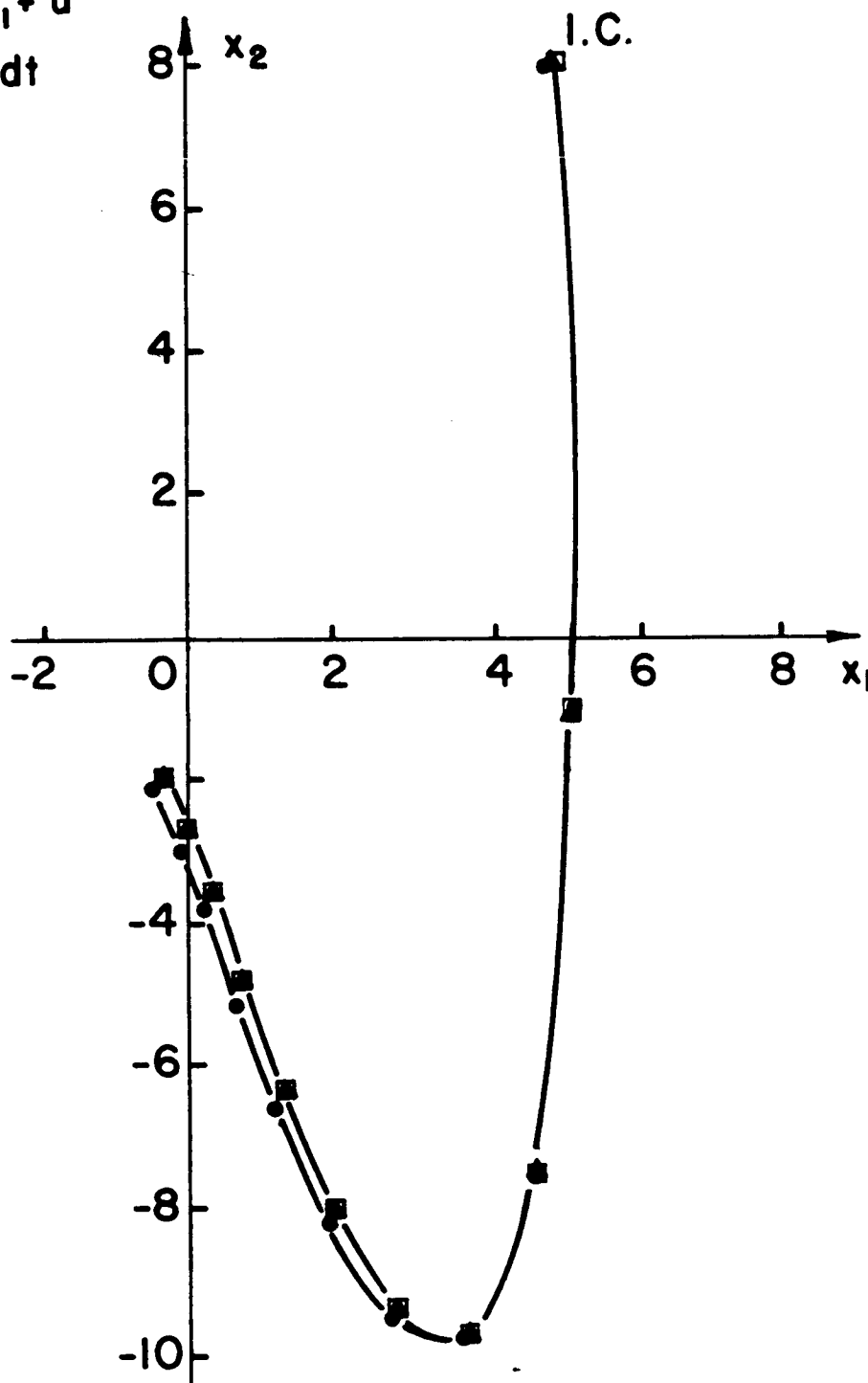
FIGURE 4.5

Comparison of trajectories for Example 4.3
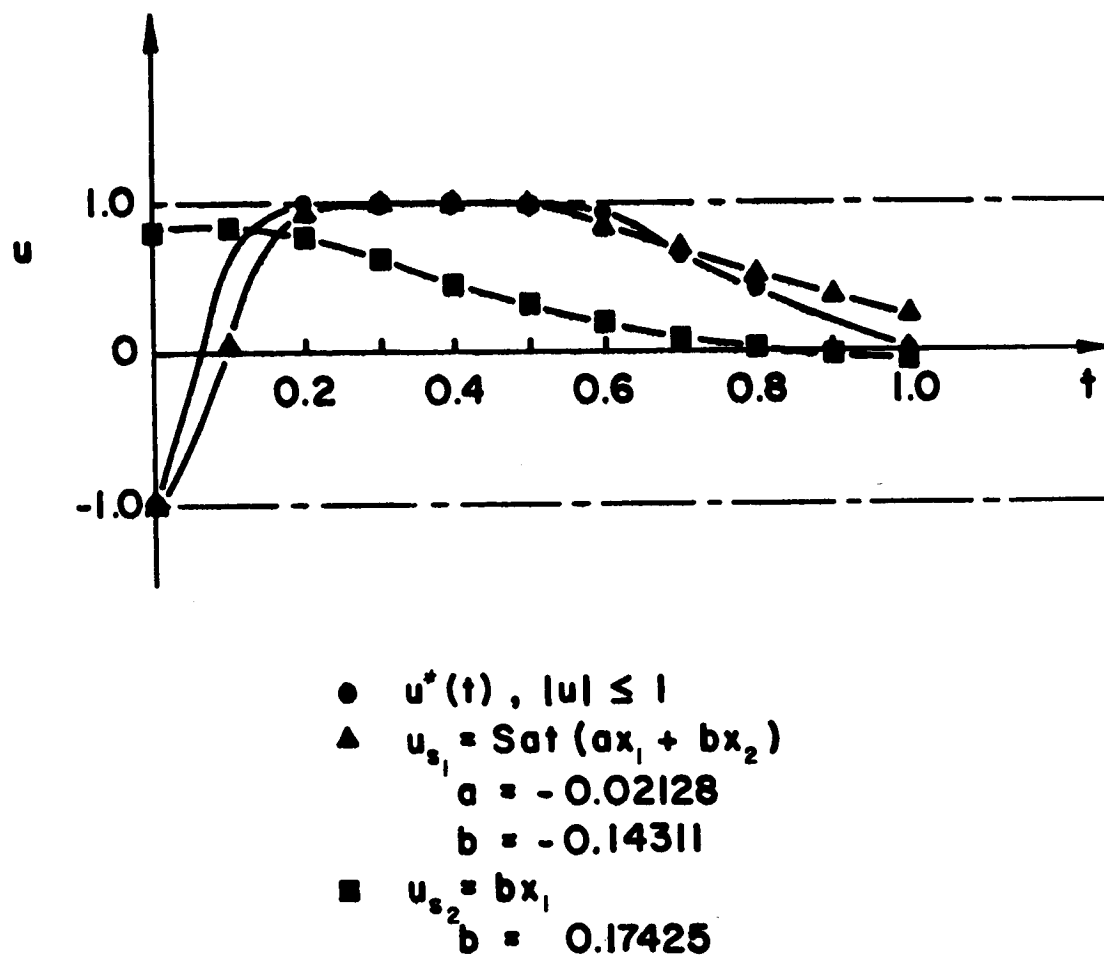
FIGURE 4.6

(ii) $u = bx_1$ along with a performance index incorporating the soft constraint of the form

$$\int_0^1 (x_1{}^2 + u_2{}^2 + u^2 + u^4)\ dt$$

P. I.   =   53.4632

b   =   0.17425

The optimal performance index is

P.I.* = 52.8735

Example 4.4

In this example consider the system of equations which governs the rotational motion of a rigid body such as a space vehicle about its center of mass. The equations of motion are

$$I_1 \dot{\omega}_1 = (I_2 - I_3)\ \omega_2\ \omega_3 + \tau_1$$

$$I_2 \dot{\omega}_2 = (I_3 - I_1)\ \omega_3\ \omega_1 + \tau_3 \qquad\qquad (4.6.7)$$

$$I_3 \dot{\omega}_3 = (I_1 - I_2)\ \omega_1\ \omega_2 + \tau_3$$

where

ω_i = angular velocity about the i-th principal axis

I_i = the moment of inertia about the i-th principal axis

τ_i = control torque on the i-th principal axis.

The constraints on the control torques are derived from the condition that the jets can only deliver a certain level of torque in the case of a space vehicle.

The constraints can be expressed in the form

$$|\tau_i| \leq T_i \qquad (4.6.8)$$

From the equations (4.6.7) and (4.6.8) the attitude control problem of a space vehicle can be posed as follows. Equation (4.6.7) represents the motion of the space vehicle just after launching (while tumbling) and (4.6.8) the constraints on the attitude control jets. The objective now is to find the appropriate torques along the principal axes such that a certain performance measure is minimized. This measure should be one that weights the angular velocities as well as the energy expended in reducing these velocities in a fixed amount of time. This amounts to slowing down the space vehicle to low angular velocities where linearizations can be made and appropriate linear controllers can take over the task of controlling the attitude accurately.

For this example, let

I_1 = 10 slug ft^2, I_2 = 20 slug ft^2 and I_3 = 40 slug ft^2

117

Then the equation (4.6.7) becomes in terms of the angular
momenta $(y_i = I_i \omega_i)$,

$$\dot{y}_1 = -0.025 y_1 y_2 + u_1$$

$$\dot{y}_2 = 0.075 y_3 y_1 + u_2$$

$$\dot{y}_3 = -0.05 \ y_1 y_2 + u_3$$

The performance criterion used will be a fixed time,
minimum energy type as given by

$$\int_0^T (y_1^2 + y_2^2 + y_3^2 + u_1^2 + u_1^2 + u_3^2) dt \qquad (4.6.10)$$

The constraints on u_1, u_2 and u_3 are taken to be

$$|u_i| \leq 1 \ , \quad i = 1, 2, 3 \qquad (4.6.11)$$

For this example, the time is fixed to be 1, and the
optimal trajectories $y_1^*(t)$, $y_2^*(t)$, $y_3^*(t)$ and the optimal
controls $u_1^*(t)$, $u_2^*(t)$, and $u_3^*(t)$ are given in Figure 4.7
for the initial conditions $y_1(0) = 10.0$, $y_2(0) = 8.0$, and
$y_3(0) = 6.0$. It is to be noted that the optimal solution
produced here is expected to act as a "design standard" when
specific controllers are designed and their relative merits
can then be evaluated.
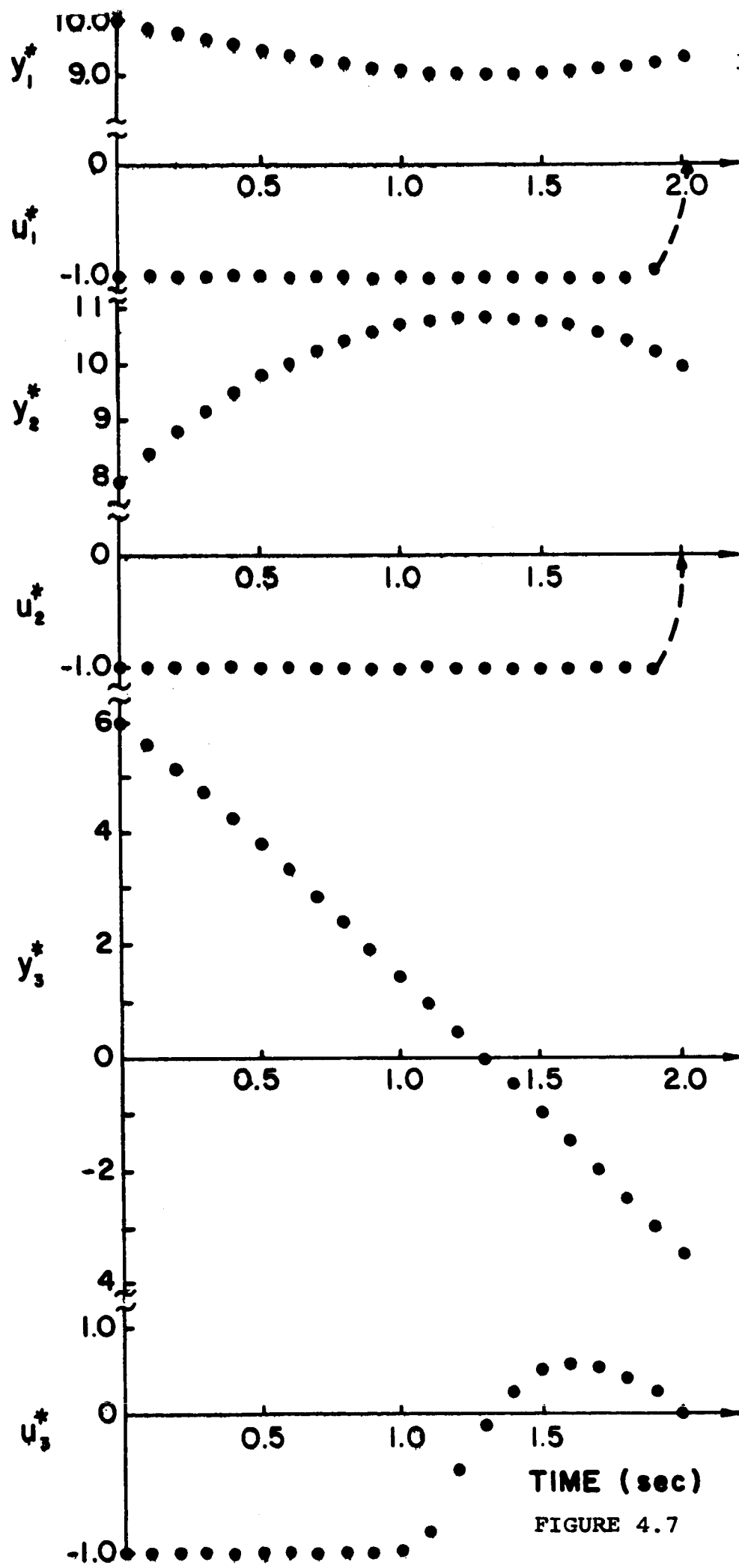
FIGURE 4.7

Remarks:

In Example 4.2 and 4.3, specific controller (i) is found by the simple gradient technique described in Chapter 3 for parameter optimization and the computational algorithm is given in Section 3.6, and the program used is listed in Appendix I. For specific controller (ii), the quasilinearization method of solving TPBVP as explained in Appendix B is utilized and the program used is listed in Appendix I. The programs for producing the optimal solutions to examples 4.2, 4.3 and 4.4 are also given in Appendix I.

4.7 Conclusions and Future Work

A computational algorithm for producing the optimal control function $\underline{u}^*(t)$ in a bounded control problem has been proposed and illustrated. The need for producing $\underline{u}^*(t)$ is for the reason of obtaining a "standard" or "yardstick" for comparison purposes. Two kinds of specific controllers have been suggested and comparisons are made with the optimal solution. The attitude control problem which is of interest to JPL has been posed and the solution by this algorithm has been presented.

Once there is a design standard, a number of specific controllers can be proposed for these problems and their

relative merits examined. The extension of this method has to be considered for free time and time optimal problems. The approach to be taken in these cases would be one of "digital experimentation" taking into account all available information from the analytical formulation of the variational problem. This means that emphasis will be given to numerical solutions of the various problems in order to produce a series of useful computational algorithms.

CHAPTER 5

STATE ESTIMATION FOR NON-LINEAR SYSTEMS

5.1 <u>Summary</u>

The problem considered in this chapter is the sequential estimation of states and parameters in noisy non-linear systems. The class of systems considered are those in which the dynamical behavior is described by an ordinary differential equation. No statistical assumptions are required concerning the nature of the unknown inputs to the system or the measurement errors on the output. For estimation purposes a least squares criterion is used. The new feature of the approach presented is that a sequential least squares estimator is obtained for the class of problems considered. This estimator could be implemented in real time. Experimental results from several examples indicate that the proposed estimation scheme is feasible.

The feasibility of using the estimated state, as produced by the sequential estimator, for control purposes is then investigated. The problem considered is the following: starting with arbitrary initial angular velocities on the

body axes of a space vehicle, synthesize control signals,
based on noisy measurements on one angular velocity, which
will force the three body angular velocities to zero. The
results of computer experiments indicate the possibility of
accomplishing angular velocity reduction in a space vehicle
using only one rate gyro.

5.2 Introduction

The sequential estimation of states and parameters in
noisy non-linear dynamical systems is of interest not only
in automatic control but also in other areas of engineering
where the system identification problem requires the process-
ing of large quantities of data.

The class of problems considered will be those in which
the dynamical behavior of the system is described by an or-
dinary differential equation. No statistical assumptions
are required concerning the nature of the input disturbances
or of the measurement errors. The absence of statistical
assumptions corresponds closely to the physical situation in
many practical problems, as the determination of valid statis-
tical data concerning disturbances is in itself a difficult
theoretical and practical problem.

The criterion that will be used for estimation is the classical least squares method. The motivation for using this criterion is historic precedent, as a least squares approach has been used explicitly and implicitly on many estimation problems with success since the time of Gauss. If valid statistical data concerning the disturbances are available then this approach will not necessarily be the best one.

The usual classical approach to least squares estimation leads to non-sequential estimation schemes. The basic objection to a non-sequential estimation scheme, when applied to a dynamical system, is that each time additional output observations are to be included, then the entire least squares calculation must be repeated. In general, the time required to perform this calculation increases with the number of measurements.

The new feature of the approach presented is that a sequential least squares estimator is obtained for the class of problems considered; this estimator could be implemented in real time.

In the formal derivation the minimization of the integral of the weighted, squared, residual errors is converted

124

to a Lagrange problem in the calculus of variations. The
Euler-Lagrange equations for this problem are written using
Pontryagin's maximum principle [21,24]. The sequential na-
ture of the estimation problem is then brought out by imbedding
the resulting two point boundary value problem (TPBVP) in a
larger class of TVBVP's using invariant imbedding techniques
[29]. A non-linear partial differential equation results
from the imbedding. Using an approximation procedure the
sequential estimator equations are derived.

The resulting estimator equations, except for an addi-
tional term, are precisely the equations obtained by Bellman,
Kagiwada, Kalaba and Sridhar [30] who consider the more re-
strictive problem in which only observation errors are allowed.
The method of derivation here is quite different from the
one used in reference [30] which is inapplicable for the
problem considered in this chapter.

In the literature the usual approach to estimation prob-
lems of this type assumes that the disturbances are gaussian
white noise of known statistics. Under these assumptions
Bryson and Frazier [31] derive a TPBVP and Cox [32] derives
a somewhat similar set of estimator equations using dynamic
programming.

A number of computer experiments were performed in order to test the feasibility of the proposed sequential least squares estimator. Experimental results are given for (1) the problem of estimating the three angular velocities of a rigid body rotating about its center of mass given noisy measurements on one angular velocity and (2) the problem of estimating position, velocity and a time varying parameter in a second order non-linear differential equation.

The feasibility of using the estimated angular velocity in example (1) above to control the space vehicle so as to reduce the angular velocities is then investigated. The results of computer experiments indicate the possibility of accomplishing angular velocity reduction in a space vehicle using only one rate gyro.

5.3 Problem Statement

The problem under consideration is that of estimating state variables and parameters in noisy non-linear dynamical systems. In this section the problem is defined for the scalar case and a physical interpretation of the proposed criterion for estimation is presented. It is a simple matter to generalize the results to the vector case; this is done in Appendix F.

Consider the class of systems defined by

$$\dot{x} = g(t, x) + k(t, x) u \qquad (5.3.1)$$

where u represents an unknown input. The explicit inclusion
of t in the right hand side of equation (5.3.1) accounts for
all known inputs. Let the output observations be denoted by

$$y(t) = h(t, x) + (\text{observation error}) \qquad (5.3.2)$$

where the (observation error) term accounts for the fact that
the output observations are of limited precision. Using the
philosophy presented in the introduction, no statistical as-
sumptions are being made concerning the unknown input or the
observation error. The estimation problem is the following:
based upon output measurements y(t) in the interval $0 \leq t \leq T$
estimate the current state x(T). A least squares criterion
will be used to estimate x(T). Using the usual least squares
terminology define the following residual errors

$$e_1(t) = y(t) - h(t, \bar{x}(t)) \qquad (5.3.3)$$

$$e_2(t) = \dot{\bar{x}} - g(t, \bar{x}(t)) \qquad (5.3.4)$$

where $\bar{x}(t)$, $0 \leq t \leq T$, represents a nominal trajectory. If
$\bar{x}(t)$ were the true trajectory of the system given by equation

(5.3.1) then for no observation errors it would follow that $e_1(t) = 0$ and for no unknown inputs it would follow that $e_2(t) = 0$.

The problem of estimating $x(T)$ in a least squares sense reduces to minimizing with respect to $\bar{x}(t)$, $0 \leq t \leq T$, the usual functional

$$\int_0^T \left[e_1^2(t) + w(t, \bar{x})\, e_2^2(t) \right] dt \qquad (5.3.5)$$

where $w(t, \bar{x})$ is a positive weighting factor. Let $\hat{x}(t)$, $0 \leq t \leq T$, denote the minimizing function; the least squares estimate of $x(T)$ is then $\hat{x}(T)$.

The estimation of $x(T)$ is then based on minimizing an integral of the sum of the weighted, squared, residual errors.

5.4 Reformulation of the Problem

It will be convenient to reformulate the problem. Substituting from equations (5.3.3) and (5.3.4) into (5.3.5) and then minimizing the expression (5.3.5) with respect to $\bar{x}(t)$, $0 \leq t \leq T$, is equivalent to minimizing

$$\int_0^T \left[\left(y - h(t, \bar{x}) \right)^2 + w(t, \bar{x})\left(\overset{\circ}{x} - g(t, \bar{x}) \right)^2 \right] dt$$

$$(5.4.1)$$

128

with respect to $\bar{x}(t)$, $0 \le t \le T$. Using equation (5.3.1) for motivation, this in turn is equivalent to minimizing

$$\int_0^T \left[\left(y - h(t, \bar{x}) \right)^2 + w(t, \bar{x}) \, k^2(t, \bar{x}) \, \bar{u}'^2 \right] dt$$

(5 4.2)

with respect to $\bar{x}(t)$ and $\bar{u}(t)$, $0 \le t \le T$, subject to the differential constraint

$$\dot{\bar{x}} = g(t, \bar{x}) + k(t, \bar{x}) \, \bar{u}$$

(5.4.3)

The minimization of the expression (5.4.2) with respect to $\bar{x}(t)$ and $\bar{u}(t)$, $0 \le t \le T$, subject to the constraint given by equation (5.4.3) constitutes the reformulation of the problem.

5.5 The Variational Problem

For the moment let the interval of observation, denoted by T, be fixed. The minimization of the expression (5.4.2) subject to equation (5.4.3) is then a Lagrange problem in the calculus of variations. The Pontryagin maximum principle [21,24] will be used to write the Euler-Lagrange equations for this variational problem.

Let

$$v(t, \bar{x}) = w(t, \bar{x}) \, k^2(t, \bar{x}) \qquad (5.5.1)$$

and define the "pre-Hamiltonian" $H(t, \bar{x}, \lambda, \bar{u})$ by

$$H(t, \bar{x}, \lambda, \bar{u}) = (y - h(t, \bar{x}))^2 + v(t, \bar{x})\bar{u}^2$$

$$+ \lambda[g(t, \bar{x}) + k(t, \bar{x}) \, \bar{u}] \qquad (5.5.2)$$

Setting $\frac{\partial H}{\partial \bar{u}} = 0$, solving for $\bar{u}(t, \bar{x}, \lambda)$ assuming $v(t, \bar{x}) \neq 0$, and substituting $\bar{u}(t, \bar{x}, \lambda)$ back into H leads to the Hamiltonian $H^*(t, x^*, \lambda)$. The variable x^* replaces $\bar{x}$ to indicate that x^* is the trajectory along which the maximum principle is satisfied. The Hamiltonian is then

$$H^*(t, x^*, \lambda) = (y - h(t, x^*))^2 + \lambda \, g(t, x^*)$$

$$\qquad (5.5.3)$$

$$- \frac{1}{4} \frac{\lambda^2}{w(t, x^*)}$$

The Euler-Lagrange equations are then

$$\dot{x}^* = \frac{\partial H^*}{\partial \lambda} \ (t, x^*, \lambda)$$

$$\qquad (5.5.4)$$

$$\dot{\lambda} = - \frac{\partial H^*}{\partial x^*} \ (t, x^*, \lambda)$$

130

since T has been fixed, and x*(0) and x*(T) are free, the
transversality conditions yield

$$\lambda(0) = 0 \qquad \lambda(T) = 0 \qquad\qquad (5.5.5)$$

Equations (5.5.4) with boundary conditions (5.5.5) is a
TPBVP. The solution of this TPBVP will yield the least
squares estimate of x(T), i.e., x*(T).

Now suppose that the observation interval is increased
to $0 \le t \le T_1$ where $T_1 > T$. In order to obtain a least
squares estimate of $x(T_1)$ using all the data observed for
$0 \le t \le T_1$ it is necessary to solve equations (5.5.4) with
boundary conditions

$$\lambda(0) = 0 \qquad \lambda(T_1) = 0 \qquad\qquad (5.5.6)$$

This is a different TPBVP than that described by equations
(5.5.4) with boundary conditions (5.5.5).

The sequential nature of the estimation problem will
now be emphasized. In the sequential problem the variable
T is regarded as an independent variable, the running time
variable. For each value of the independent time variable
T, in order to estimate the current state x(T) in a least
squares sense using all the observations in interval 0 to T,

it is necessary to solve a TPBVP of the above type. The sequential nature of the estimation problem then leads naturally to the use of invariant imbedding techniques [29].

5.6 The Invariant Imbedding Equations

In order to solve the sequential estimation problem it is necessary to solve the TPBVP described by equations (5.5.4) with boundary conditions

$$\lambda(0) = 0 \qquad \lambda(T) = 0 \qquad (5.6.1)$$

for all values of the variable T, the running time variable.

Using the techniques of invariant imbedding [29] replace the boundary conditions (5.6.1) by the more general conditions

$$\lambda(0) = 0 \qquad \lambda(T) = C \qquad (5.6.2)$$

Let $r(C, T)$ be the missing terminal condition on x^* given that x^* and λ satisfy the TPBVP described by equations (5.5.4) with boundary conditions (5.6.2). It can be shown [Appendix D] that $r(C, T)$ satisfies

$$\frac{\partial r}{\partial T} - \frac{\partial r}{\partial C} \frac{\partial H^*}{\partial r} (T, r, C) = \frac{\partial H^*}{\partial C} (T, r, C) \qquad (5.6.3)$$

132

The solution to the non-linear partial differential equation (5.6.3) with the proper boundary conditions on r contains the solution to all TPBVP's consisting of equations (5.5.4) with boundary conditions given by equations (5.6.2). In order to solve the sequential least squares estimation problem it is necessary to determine $r(0, T)$ since $\lambda(T) = 0$.

5.7 The Sequential Estimator Results

The partial differential equation (5.6.3) may be transformed approximately into an initial value problem by substituting $r(C, T) = -P(T) C + \hat{x}(T)$ and expanding about $r(0, T)$ retaining terms to first order in C. The motivation for this approach is that only those solutions of equation (5.6.3) for which $C = 0$ are of interest. Also the least squares estimate of $x(T)$, now denoted by $\hat{x}(T)$ to emphasize the sequential nature of the problem, is $r(0, T)$. The results [Appendix E] are

$$\frac{d\hat{x}}{dT} = g(T, \hat{x}) + 2P(T) h_{\hat{x}}(T, \hat{x}) [y - h(T, \hat{x})]$$

$$\frac{dP}{dT} = 2P(T) g_{\hat{x}}(T, \hat{x}) \tag{5.7.1}$$

$$+ 2P \frac{\partial}{\partial \hat{x}} \{h_{\hat{x}}(T, \hat{x}) [y - h(T, \hat{x}]\}$$

$$+ \frac{1}{2w(T, \hat{x})}$$

where $h_{\hat{x}} = \dfrac{\partial h}{\partial \hat{x}} (T, \hat{x})$

The differential equations (5.7.1) are the principal result; they describe a filter which operates in real time on the observations to sequentially produce least squares estimates of the current state.

Comparing the equation for P with that obtained by Bellman, Kagiwada, Kalaba and Sridhar [30] for the more restrictive problem which allows observation errors only, it is interesting to note that the $\left(\dfrac{1}{2w(T, \hat{x})}\right)$ term is the only modification necessary to account for unknown inputs.

The results for the vector case are given in Appendix F.

5.8 Experimental Results - Estimation

a) Procedure

A number of controlled, computer experiments were performed in order to test the feasibility of the proposed sequential least squares estimator. Each experiment was divided into two phases. In phase 1 the system trajectory was generated by solving equations (5.3.1). In phase 2 the output data from the system was corrupted with measurement noise, i.e., y(t) from equation (5.3.2) was generated, and finally the noisy observations were used as an input to the sequential estimator as described by equations (5.7.1).

134

The model used for the measurements was

$$y(t) = P_1 \cdot r_1(t) \cdot |x(t)| + x(t) + P_2 \cdot r_2(t) \quad (5.8.1)$$

where $x(t)$ — the variable measured

     $y(t)$ — the observed value of $x(t)$

$r_1(t), r_2(t)$ — for each t, statistically independent random variables, uniformly distributed between -1 and $+1$.

    P_1, P_2 — constants, used to adjust the relative magnitude of the error.

An interpretation of the model for the measurements is as follows: suppose the maximum magnitude of $x(t)$ is in the order of unity, then with $P_1 = P_2 = 0.1$ the error model corresponds to measurements accurate, on the average, to approximately one significant figure. Therefore if the magnitude of $x(t)$ is approximately known the relative accuracy of the measurements may be controlled by adjusting P_1 and P_2.

b) Example – Rotational Motion of a Rigid Body About Its Center of Mass

The following question provides the physical motivation for this example: Is it possible to sequentially estimate the three angular velocities about the principal body axis of a rotating body given noisy measurements on only one angular velocity? The equations of motion are

$$\dot{\omega}_1 = \frac{I_2 - I_3}{I_1} \omega_2 \omega_3 + u_1 \qquad (5.8.2)$$

$$\dot{\omega}_2 = \frac{I_3 - I_1}{I_2} \omega_1 \ \omega_3 + u_2$$

$$\dot{\omega}_3 = \frac{I_1 - I_2}{I_3} \omega_1 \ \omega_2 + u_3$$

where

ω_i = angular velocity about the ith principal axis

u_i = disturbance torque/I_i for the ith principal axis

I_i = moment of inertia about the ith principal axis

Let

$I_1 = 10$ slug ft^2, $I_2 = 20$ slug ft^2 and $I_3 = 40$ slug ft^2

(i) Consider first the case when it is known that there are no disturbance inputs acting on the system, i.e. $u_1 = u_2 = u_3 = 0$ in equation (5.8.2). Using the results in Appendix F, equation (F.31) the sequential estimator equations for this example become

$$\dot{\hat{\omega}}_1 = -2 \ \hat{\omega}_2 \ \hat{\omega}_3 + 2 \ P_{11}(\tilde{\omega}_1 - \hat{\omega}_1)$$

$$\dot{\hat{\omega}}_2 = 1.5 \ \hat{\omega}_1 \ \hat{\omega}_3 + 2 \ P_{21}(\tilde{\omega}_1 - \hat{\omega}_1) \qquad (5.8.3)$$

$$\dot{\hat{\omega}}_3 = -0.25 \ \hat{\omega}_1 \ \hat{\omega}_2 + 2P_{31}(\tilde{\omega}_1 - \hat{\omega}_1)$$

$$\dot{P} = -2PHQH' \ P + g_{\hat{\omega}} \ P + P \ g'_{\hat{\omega}}$$

where

$$Q = 1$$

$$P = \begin{bmatrix} P_{11} & P_{12} & P_{13} \\ P_{21} & P_{22} & P_{23} \\ P_{31} & P_{32} & P_{33} \end{bmatrix} \qquad HQH' = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$g_{\hat{\omega}} = \left[\frac{\partial g_i}{\partial \hat{\omega}_j} \right] = \begin{bmatrix} 0 & -2\hat{\omega}_3 & -2\hat{\omega}_2 \\ 1.5\hat{\omega}_3 & 0 & 1.5\hat{\omega}_1 \\ -0.25\hat{\omega}_2 & -0.25\hat{\omega}_1 & 0 \end{bmatrix}$$

$\tilde{\omega}_1$ = measured value of ω_1 using the error model given in equation (5.8.1), with $P_1 = 0.1$ and $P_2 = 0.1$.

$'$ - denotes transpose

Figure 5.1 displays the results for $\hat{\omega}_i$ i = 1, 2, 3

using initial conditions

$$\hat{\omega}(0) = \begin{bmatrix} 0.9 \\ 0.0 \\ 0.0 \end{bmatrix} \qquad\qquad (5.8.4)$$

$$P(0) = \begin{bmatrix} 3 & 1 & 3 \\ 1 & 3 & 1 \\ 1 & 1 & 3 \end{bmatrix}$$

The initial conditions for $\hat{\omega}_i$ reflect the physical situation, i.e., for $\hat{\omega}_1(0)$ is used $\tilde{\omega}_1(0-)$ whereas zero is selected for $\hat{\omega}_2(0)$ and $\hat{\omega}_3(0)$ as no information is available. The on-diagonal terms in P(0) reflect in some manner the confidence one has in the initial values of $\hat{\omega}_i$.

ii) In the case of disturbance inputs to the system, referring to Appendix F, let $W(t, \hat{x}) = \frac{1}{2} I$ where I = identity matrix. Then, since $k(t, \hat{x}) = I$, the only modification to the sequential estimator equations (5.8.3) is to add the identity matrix to the right hand side of the P equations.

Figure 5.2 displays the results for $\hat{\omega}_i$ i = 1, 2, 3 with constant disturbance inputs $u_1 = 0.005$, $u_2 = -0.005$, $u_3 = 0.005$ acting on the system.

Figure 5.3 displays the results for $\hat{\omega}_i$ i = 1, 2, 3 with random disturbance inputs acting on the system. The model for the disturbance inputs was

$$u_i(t) = 0.01 \quad r_i(t) \qquad i = 1, 2, 3 \qquad (5.8.5)$$

where

$r_i(t)$ i = 1, 2, 3 for each t, are statistically independent random variables uniformly distributed between -1 and +1.

ESTIMATION OF STATE VARIABLES
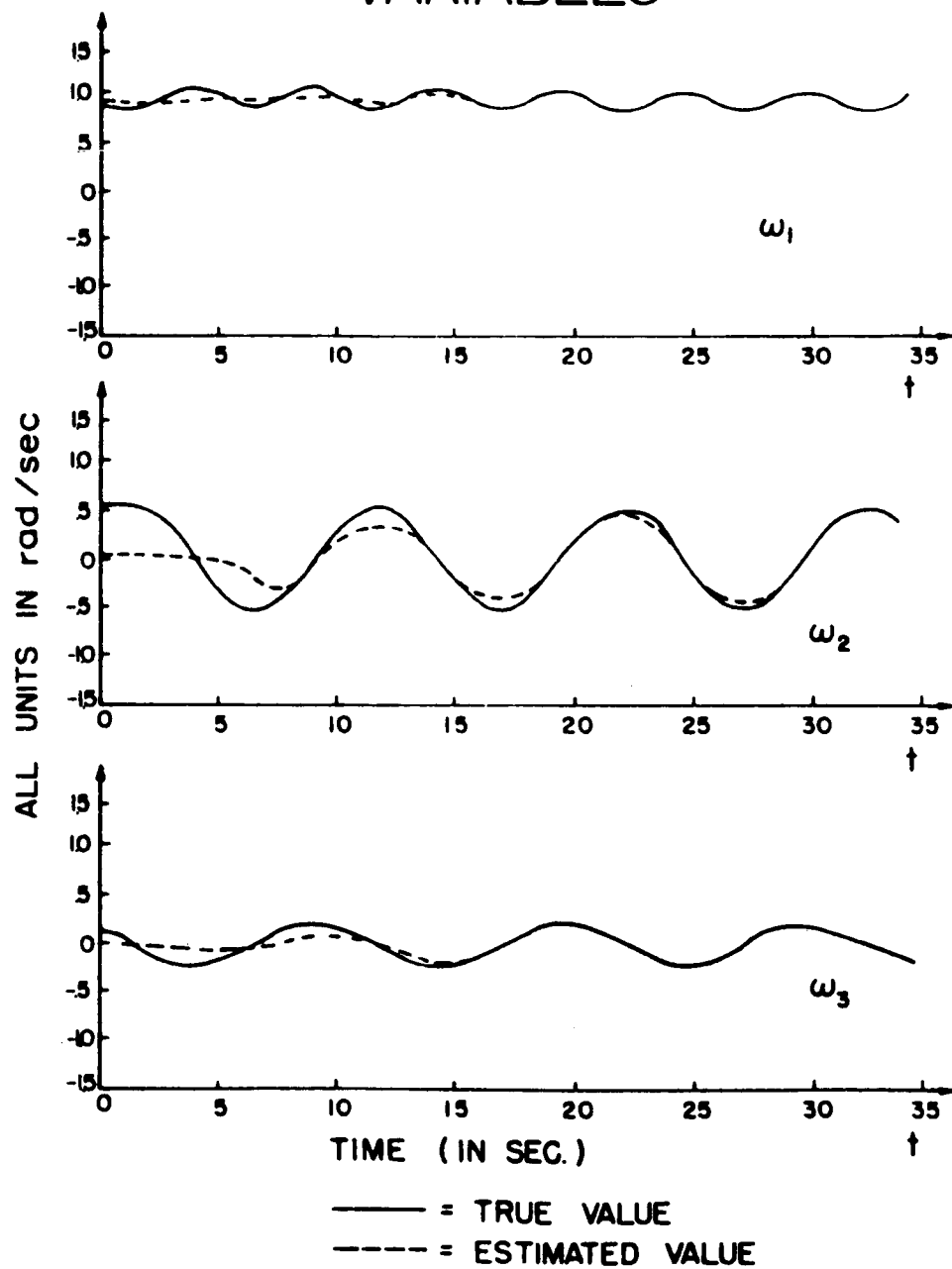
FIGURE 5.1

ESTIMATION OF STATE VARIABLES - CONSTANT DISTURBANCE TORQUES ACTING ON VEHICLE
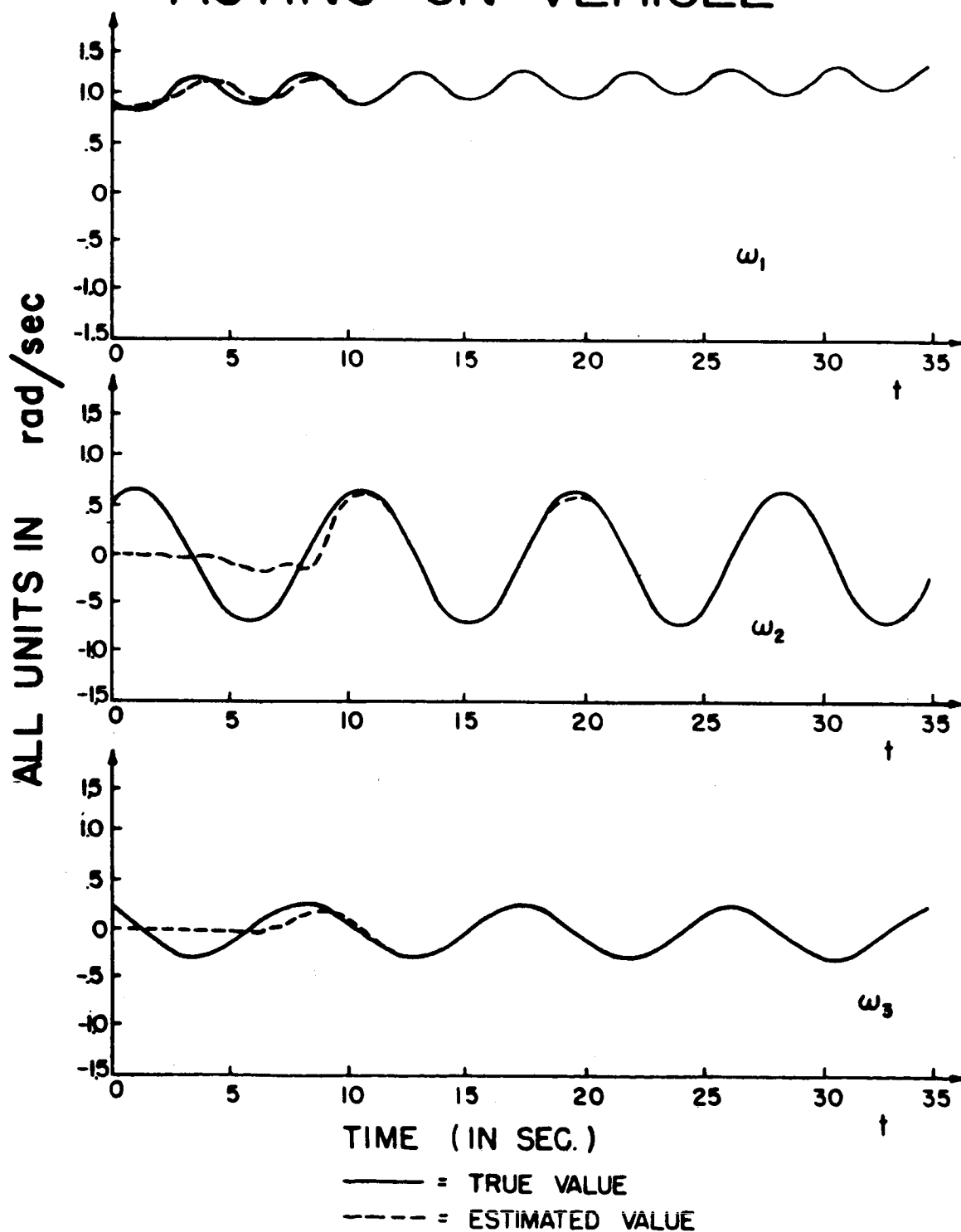
FIGURE 5.2

ESTIMATION OF STATE VARIABLES - RANDOM DISTURBANCE TORQUES ACTING ON VEHICLE
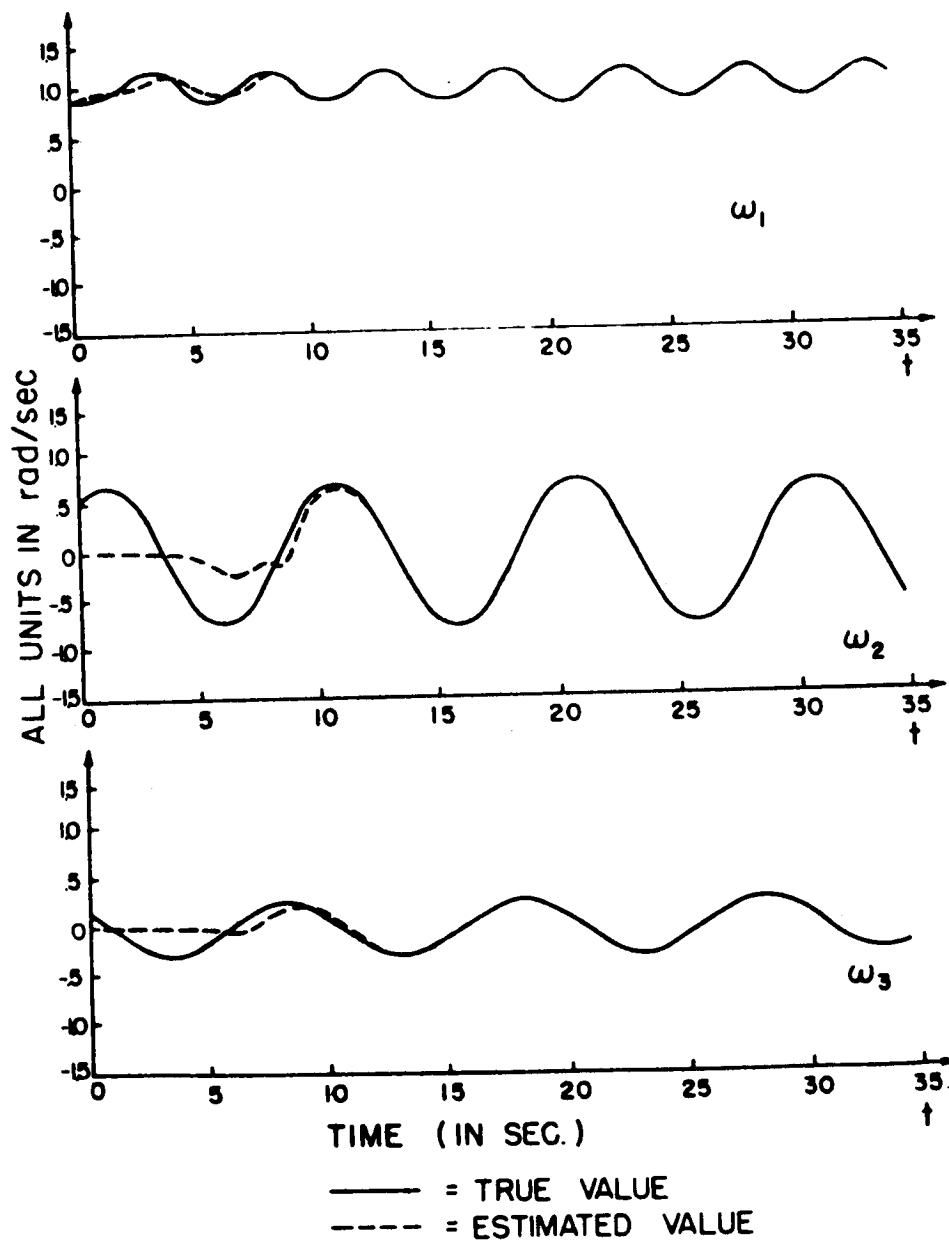
FIGURE 5.3

While the disturbance inputs used in these two examples may seem to be of low level, it has been determined experimentally that properly applied inputs in the order of 0.05 on each axis will reduce the ω_i's to zero in 15 to 20 seconds.

The estimates of all the ω_i's were within 5% of their true values at the end of 34 seconds. Some estimates were within 0.2% of their true values.

Comparing Figures 5.3 and 5.2 with Figure 5.1 it is interesting to note that the additional term in the P equations, which accounts for unknown inputs, resulted in the state estimator "tracking" the angular velocities ω_i i = 1,2,3 considerably earlier than in the case when this term was not present. Due to this experimental evidence the additional term in the P equations was retained for the remaining examples.

c) Example - <u>Estimation of Position, Velocity and a</u>
   <u>Time Varying Parameter</u>

Since the method presented makes no distinction between state variables and unknown parameters which may be modeled by differential equations. the following example is quite pertinent to the type of problems of interest to JPL even though parameters are also being estimated here.

The system equations are

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = -2x_1 - a(t)x_1^3 - 3x_2 + 5 \sin (t) \qquad (5.8.6)$$

where

$$a(t) = 2e^{-0.1t} \qquad (5.8.7)$$

and the output observations are

$$y(t) = \text{observed value of } x_1(t) \qquad (5.8.8)$$

using the model given by equation (5.8.1) with $P_1 = P_2 = 0.1$.

Suppose the form of $a(t)$ is known but not its initial value or "time constant", then $a(t)$ may be modeled by

$$\dot{a}(t) = -b\ a(t)$$
$$\qquad\qquad\qquad (5.8.9)$$
$$\dot{b} = 0$$

where the initial conditions on b and $a(t)$ are unknown.

The sequential estimation problem is, based on $y(t)$ $0 \le t \le T$, to estimate $x_1(T)$, $x_2(T)$, $a(T)$, and $b(T) = b(0)$. Define $x_3(t) = a(t)$ and $x_4(t) = b$. Using the results in Appendix F, the estimator equations for this example become

$$\dot{\hat{x}}_1 = \hat{x}_2 + 2P_{11}(y - \hat{x}_1)$$

$$\dot{\hat{x}}_2 = -2\hat{x}_1 - \hat{x}_3\hat{x}_1{}^3 - 3\hat{x}_2 + 5 \sin (t) + 2P_{21}(y - \hat{x}_1)$$

$$\dot{\hat{x}}_3 = -\hat{x}_4\hat{x}_3 + 2P_{31}(y - \hat{x}_1) \tag{5.8.10}$$

$$\dot{\hat{x}}_4 = 2P_{41}(y - \hat{x}_1)$$

$$\dot{P} = -2PHQH'P + g_{\hat{x}}P + Pg_{\hat{x}}' + I$$

where

$$Q = 1$$

$$P = [P_{ij}] \text{ is a } 4 \times 4 \text{ matrix}$$

$$HQH' = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & & & \cdot \\ 0 & & & \cdot \\ 0 & \cdot & \cdot & \cdot & 0 \end{bmatrix}$$

$$g_{\hat{x}} = \left[\frac{\partial g_i}{\partial \hat{x}_j}\right] = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -2 & -3\hat{x}_3\hat{x}_1{}^2 -3 & -x_1{}^3 & 0 \\ 0 & 0 & -\hat{x}_4 & -\hat{x}_3 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

144

Figure 5.4 displays the results for $\hat{x}_i$ $i = 1, 2, 3, 4$

obtained with initial conditions for the estimator of

$$\hat{x}(0) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

(5.8.11)

$$P(0) = \begin{bmatrix} 3 & 1 & 1 & 1 \\ 1 & 3 & 1 & 1 \\ 1 & 1 & 3 & 1 \\ 1 & 1 & 1 & 3 \end{bmatrix}$$

It is interesting to note that the estimator "tracks" x_1 and x_2 considerably sooner than it "tracks" either $a(t)$ or b. Apparently the coupling between the $\hat{x}$ and P equations compensates for the initially poor estimates on $a(t)$ and b.

d)   Example - Rotational Motion of a Rigid Body, Linear Combination of Angular Velocities Measured

The examples presented here are similar, with two exceptions, to those in part (a).

The major difference is that here the output of the system will be assumed to be a linear combination of the three body angular velocities. Outputs of this type could be obtained by using one rate gyro which is skewed with

ESTIMATION OF POSITION VELOCITY AND TIME VARYING PARAMETER

FIGURE 5.4

146
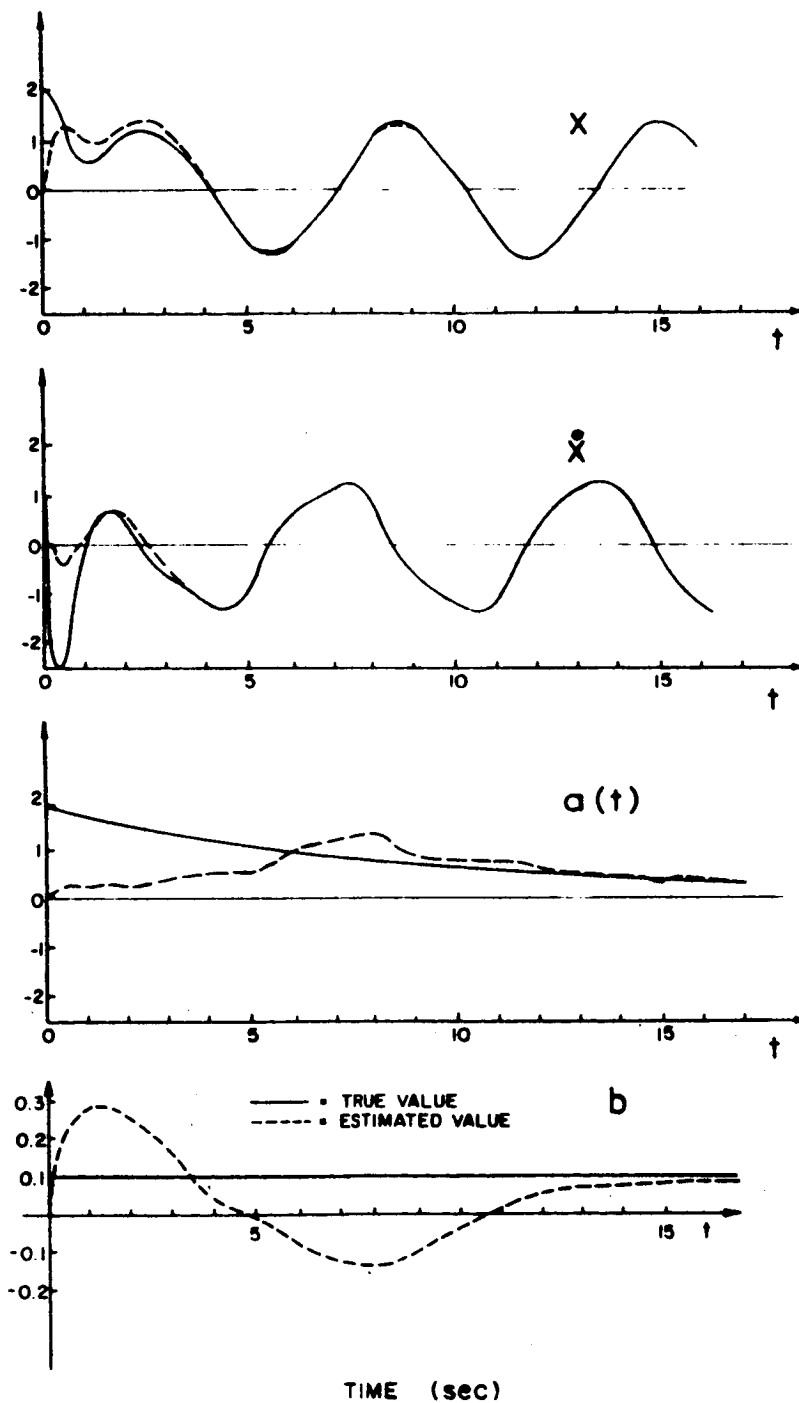
respect to the three principal body axes. The motivation
for measuring a linear combination of the angular velocities
about the principal body axes, is that with measurements of
only one angular velocity about a principal axis there is
not a unique solution to the state estimation problem in
the untorqued case. The equations of motion in the untorqued
case are

$$\dot{\omega}_1 = c_1 \, \omega_2 \, \omega_3$$

$$\dot{\omega}_2 = c_2 \, \omega_1 \, \omega_3 \qquad\qquad (5.8.12)$$

$$\dot{\omega}_3 = c_3 \, \omega_1 \, \omega_2$$

where c_i $i = 1, 2, 3$ are known constants. Let $X_1(t)$, $X_2(t)$,
$X_3(t)$ represent a solution of equations (5.8.12), i.e.
$\omega_i(t) = X_i(t)$ $i = 1, 2, 3$, then $X_1(t)$, $-X_2(t)$, $-X_3(t)$ also
represents a solution of equations (5.8.12). Hence if only
the angular velocity $\omega_1(t)$ is measured it is not possible
in the untorqued case to distinguish between the above two
solutions. This ambiguity in sign does not exist if a linear
combination of the ω_i $i = 1, 2, 3$ is measured.

The second difference is that the numbers used for the
principal body moments of inertia and initial conditions on
the body rates are comparable to the problems of interest to
JPL.

The system equations are

$$\dot{\omega}_1 = \frac{I_2 - I_3}{I_1} \omega_2 \ \omega_3$$

$$\dot{\omega}_2 = \frac{I_3 - I_1}{I_2} \omega_1 \ \omega_3 \qquad (5.8.13)$$

$$\dot{\omega}_3 = \frac{I_1 - I_2}{I_3} \omega_2 \ \omega_1$$

and the output is

$$(5.8.14)$$

$$y = h_1 \omega_1 + h_2 \omega_2 + h_3 \omega_3 + (\text{observation error})$$

where the model for the measurements is given by equation

(5.8.1). Let $I_1 = 92$ slug ft^2, $I_2 = 113$ slug ft^2, $I_3 = 63$

slug ft^2, and $h_1 = h_2 = h_3 = 0.578$. The values of the h_i's

were arbitrarily selected so that each angular velocity about

a principal axis was weighted equally in the output. Using

the results in Appendix F, equation (F.31), the sequential

estimator equations for this example become

$$\dot{\hat{\omega}}_1 = 0.576 \ \hat{\omega}_2 \ \hat{\omega}_3 + 2 \left(\sum_{i=1}^{3} P_{1i} \ h_i \right) \cdot z$$

$$\dot{\hat{\omega}}_2 = -0.283 \ \hat{\omega}_1 \ \hat{\omega}_3 + 2 \left(\sum_{i=1}^{3} P_{2i} \ h_i \right) \cdot z \qquad (5.8.15)$$

$$\dot{\hat{\omega}}_3 = -0.35 \; \hat{\omega}_1 \quad \hat{\omega}_2 \; + \; 2\left(\sum_{i=1}^{3} P_{3i} \; h_i \right) \cdot z$$

$$\dot{P} = -2P \; HQH'P + g_{\hat{\omega}} P + Pg_{\hat{\omega}}' + 0.002 \cdot I$$

where

$$Z = y - h_1 \hat{\omega}_1 - h_2 \hat{\omega}_2 - h_3 \hat{\omega}_3$$

$$P = 3 \times 3 \text{ matrix}$$

$$Q = 1$$

$$HQH' = \{h_i \; h_j\}$$

$$g_{\hat{\omega}} = \begin{bmatrix} 0 & 0.576 \; \hat{\omega}_3 & 0.576 \; \hat{\omega}_2 \\ -0.283 \; \hat{\omega}_3 & 0 & -0.283 \; \hat{\omega}_1 \\ -0.35 \; \hat{\omega}_2 & -0.35 \; \hat{\omega}_1 & 0 \end{bmatrix}$$

Figures 5.5 and 5.6 display the results for two different sets of initial conditions for the system equations (5.8.13). Also included on these graphs are the true output, the estimated output, and some of the measured outputs. The measured values are shown at ten second intervals in order to convey some feeling for the type of measurement errors given by the model used. In both of these examples the parameters used for the noise model, given by equation (5.8.1), were $P_1 = 0.1$

ESTIMATION OF STATE VARIABLES

FIGURE 5.5

ESTIMATION OF STATE VARIABLES
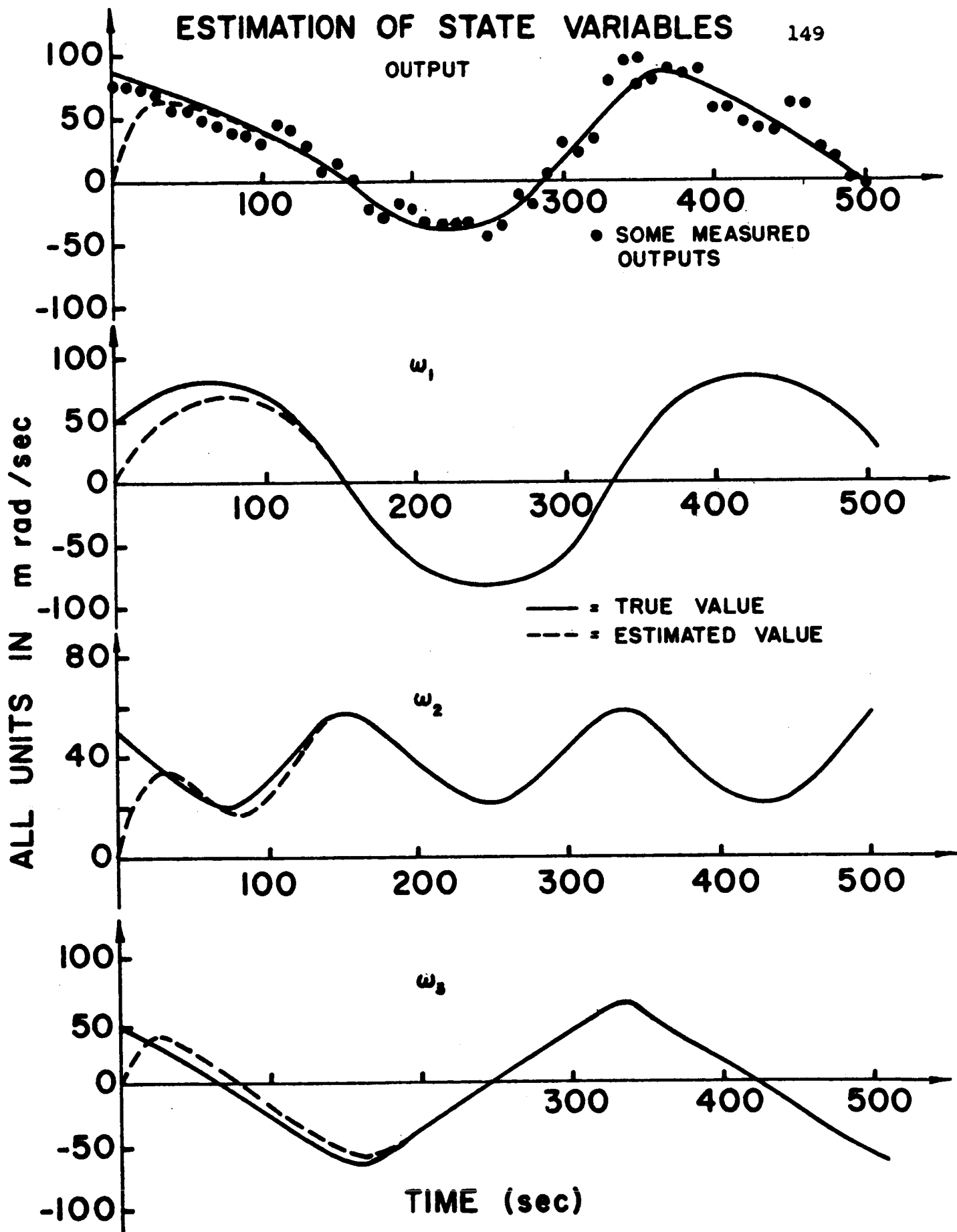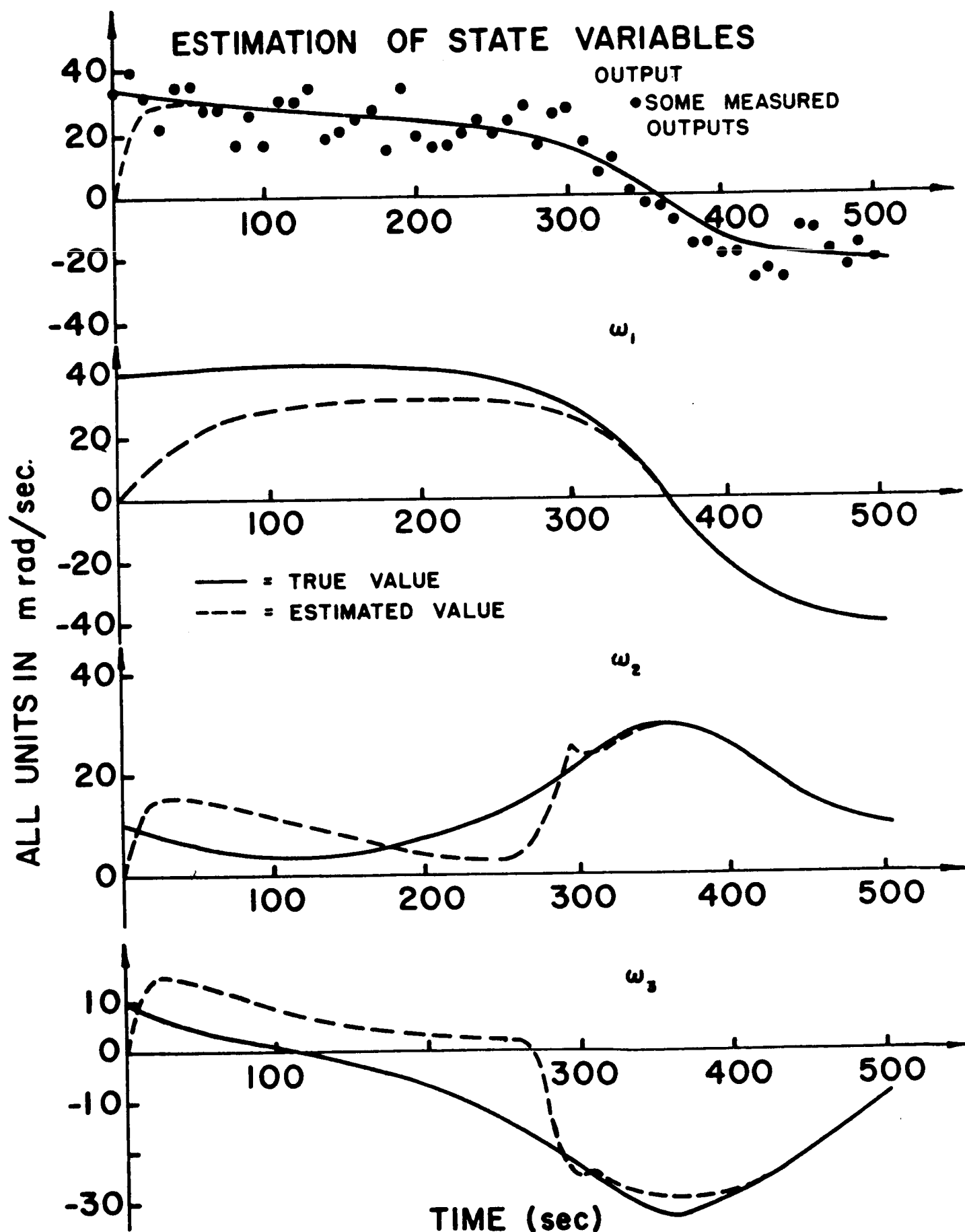
FIGURE 5.6

and $P_2 = 0.01$. The initial conditions used for the estimator equations were

$$\hat{\omega}(0) = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\tag{5.8.16}$$

$$P(0) = \begin{bmatrix} 0.03 & 0.01 & 0.01 \\ 0.01 & 0.03 & 0.01 \\ 0.01 & 0.01 & 0.03 \end{bmatrix}$$

The initial values for the estimates $\hat{\omega}_i$ $i = 1, 2, 3$ reflect the fact that since a linear combination of the angular velocities is being measured, no information concerning the individual velocities is available at time $t = 0$.

5.9 Control Using Estimated State

The above section has shown that the proposed estimation scheme provides a feasible method of sequential estimation of state variables in noisy non-linear dynamical systems. The next question is then, is it possible to use the estimated state variables for control purposes? Of interest, to JPL, specifically: Is it possible to use the estimated angular velocities about the three body axes for the attitude

control of a space vehicle? This section will consider the feasibility of using the estimated angular velocities, as produced by the non-linear state estimator, for rate reduction purposes. The motivation for the approach presented is the desire to remove two rate gyros from a space vehicle attitude control system.

The problem considered is then, that of using noisy measurements on one angular velocity to sequentially estimate all three angular velocities about the principal body axes and then to use these estimated velocities for rate reduction. Figure 5.7 displays the overall block diagram of the system used for the initial feasibility studies. Since the first portion of the overall control scheme is concerned with state variable estimation, the switches indicate that at time t = T the controller is turned on and then the estimated state is used for control purposes.

The controller selected for these experiments represents a "bang-bang" type of controller which could be physically realized by the use of on-off gas jets. Actually the portion of the overall system contained within the dashed lines is rather arbitrarily labeled the controller. The controller could just as easily be thought of as the non-linear

CONTROL USING ESTIMATED STATE
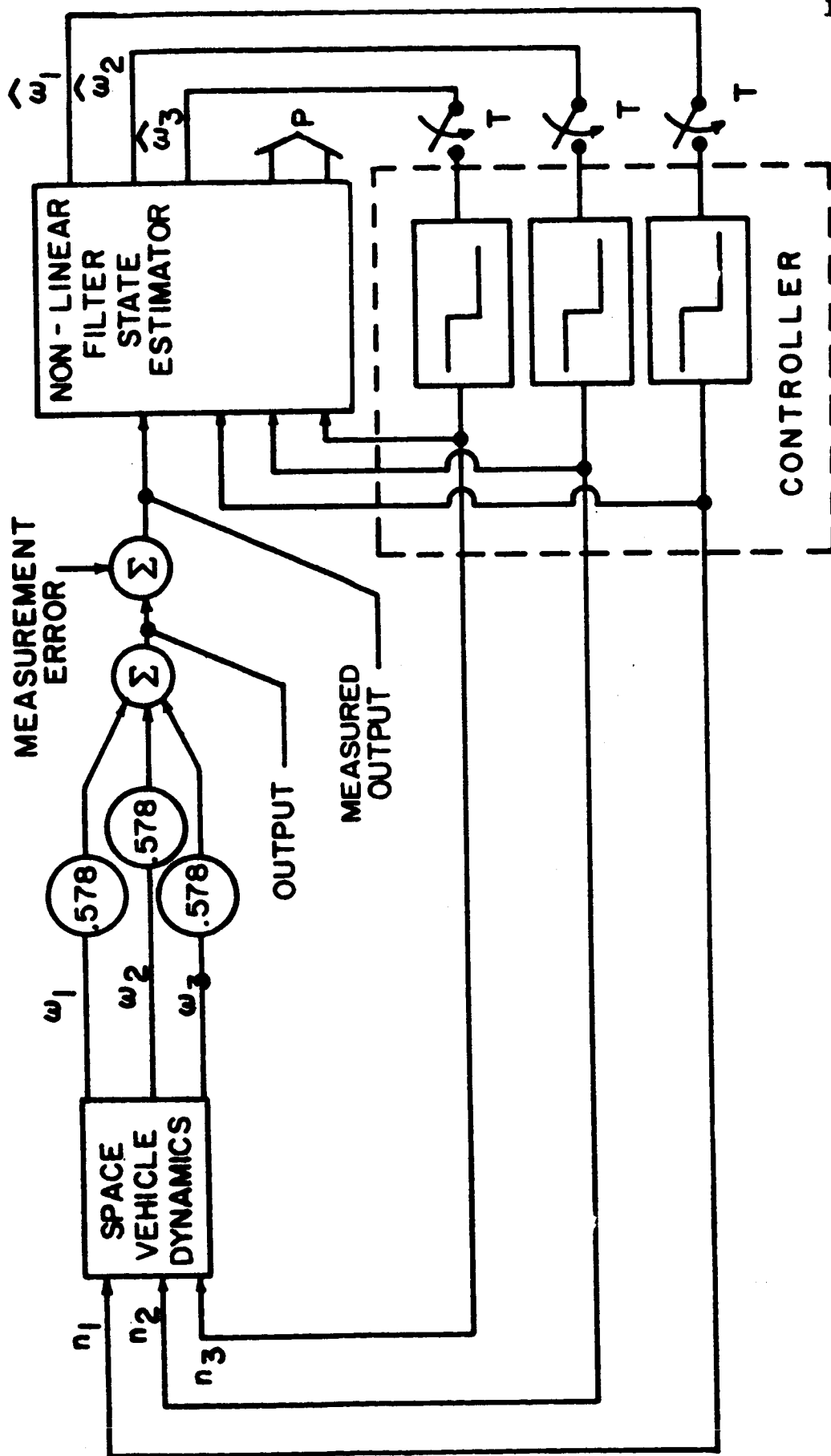
CONSTANT = 0.0005 rad/(sec.)^2

FIGURE 5.7

ACCELERATION

153

154

state estimator plus the relays, as this system receives as an input the measured output of the plant and processes this data to produce signals which will control the plant.

5.10 Experimental Results - Control Using Estimated State

A number of computer experiments were performed in order to test the feasibility of the proposed method for rate reduction in the space vehicle attitude control problem. In each experiment the non-linear state estimator was allowed to run, without the controller, for 260 seconds, at which time the controller was turned on.

The principal moments of inertia and acceleration constants used were comparable to problems of interest to JPL. The values for the principal moments of inertia were identical to those used in the state estimation examples (part d). The acceleration constants used were 0.0005 rad/(sec)^2 on all three axes. Also in all the experiments the initial conditions for the estimator equations were the same as in estimation - only experiments, i.e. equations (5.8.16).

Figure 5.8 displays the results for the plant initial conditions of $\omega_1 = \omega_2 = \omega_3 = 50$ milliradians/sec. The parameters used for the measurement model, given by equation (5.8.1), were $P_1 = 0.1$ and $P_2 = 0.01$.

CONTROL USING ESTIMATED STATE

FIGURE 5.8

CONTROL USING ESTIMATED STATE

FIGURE 5.9

CONTROL USING ESTIMATED STATE
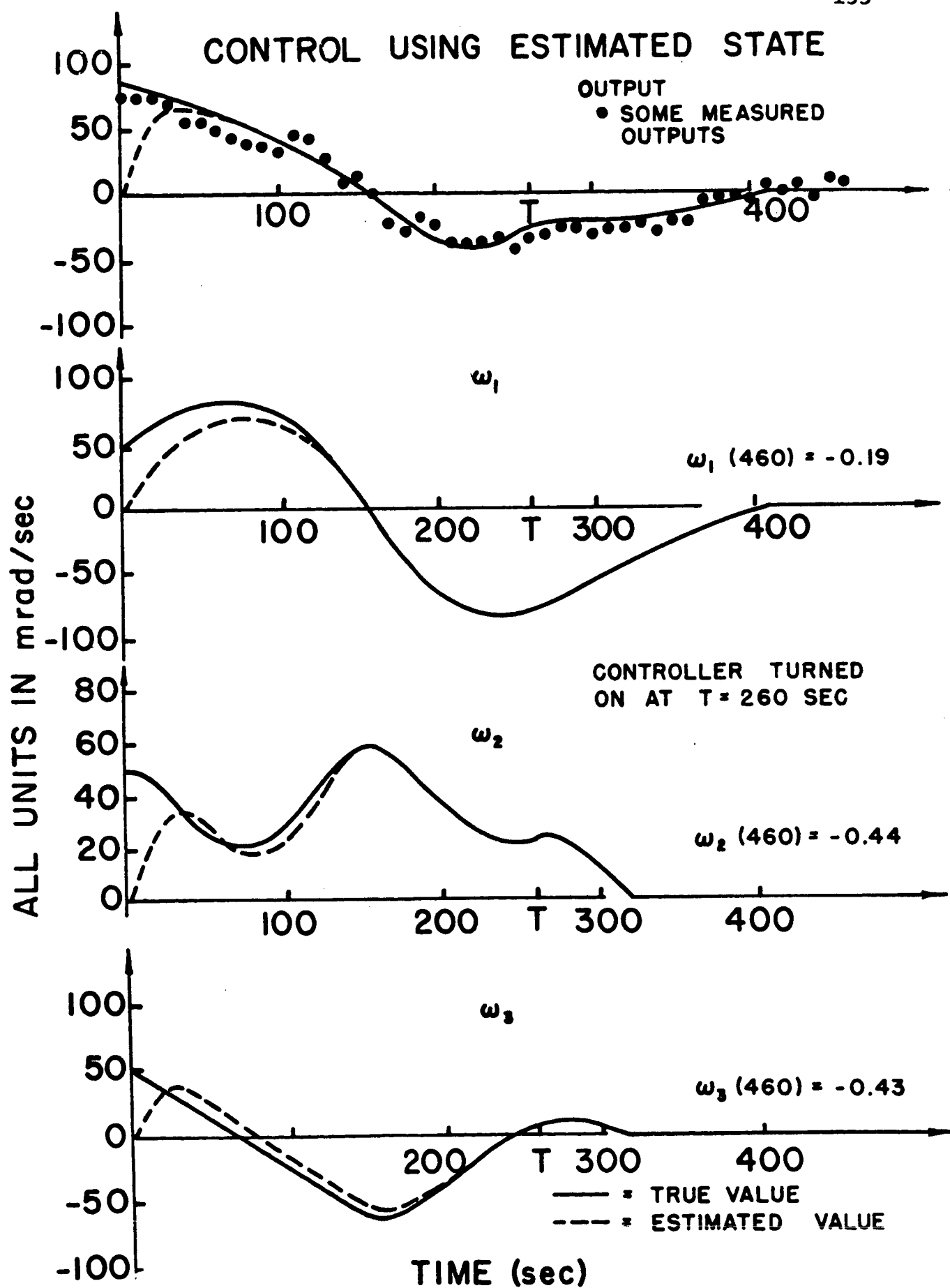
FIGURE 5.10

CONTROL USING ESTIMATED STATE

FIGURE 5.11

CONTROL USING ESTIMATATED STATE

OUTPUT

• SOME MEASURED
OUTPUTS

ω_1

$\omega_1 (660) = 0.9$

CONTROLLER TURNED
ON AT T = 260 sec,
5% ERROR IN
CONTROL TORQUES

ω_2

$\omega_2 (660) = -0.2$

——— = TRUE VALUE

— — — = ESTIMATED VALUE

ω_3

$\omega_3 (660) = -0.7$

TIME (sec)

ALL UNITS IN m rad/sec.
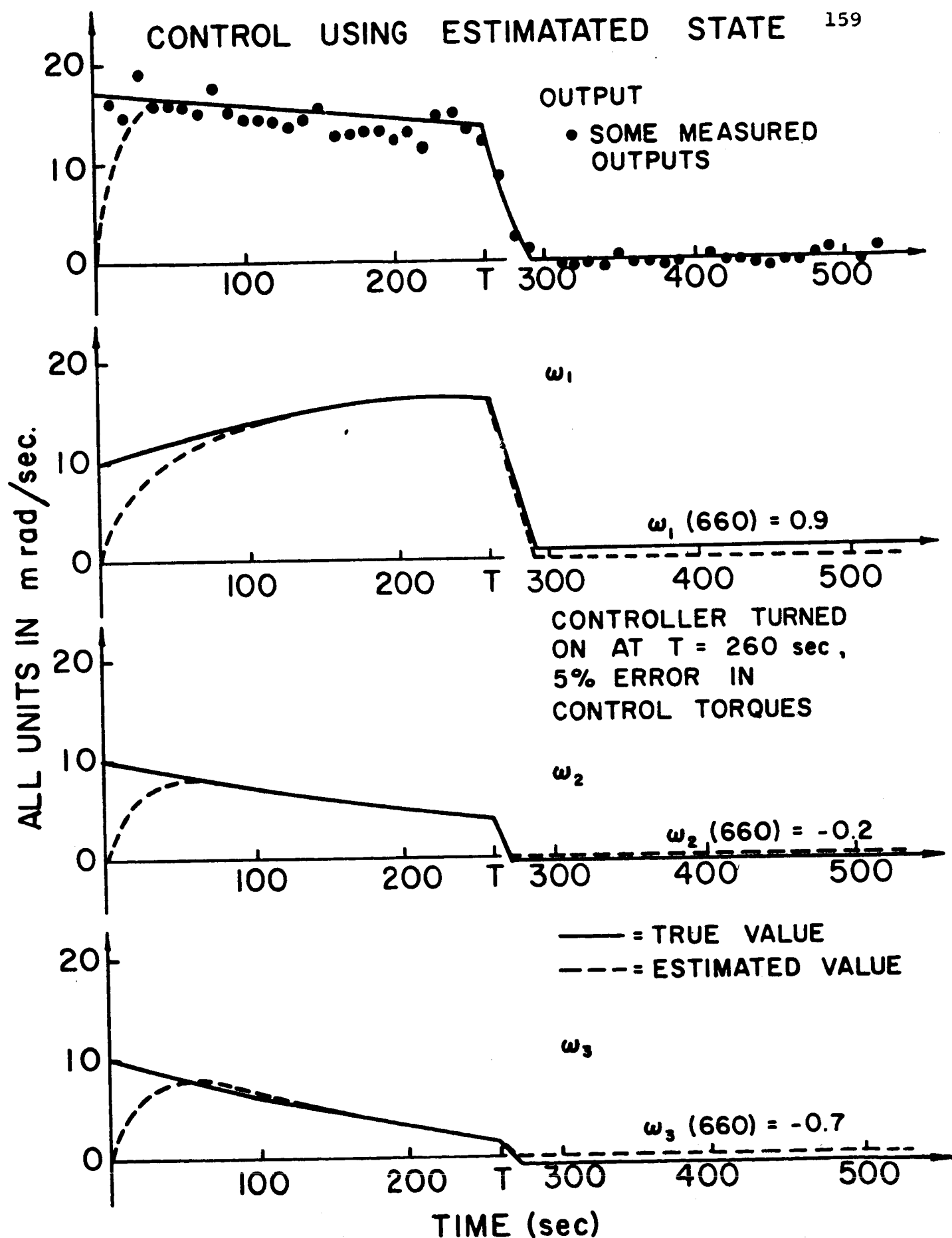
FIGURE 5.12

Figure 5.9 displays the results for the plant initial conditions of $\omega_1 = \omega_2 = \omega_3 = 10$ mrad/sec. The parameters in the error model were $P_1 = 0.1$ and $P_2 = 0.001$.

Figure 5.10 displays the results for the plant initial conditions of $\omega_1 = 100$ mrad/sec, $\omega_2 = \omega_3 = 50$ mrad/sec. The parameters in the error model were $P_1 = 0.1$ and $P_2 = 0.01$.

Figures 5.11 and 5.12 display the results for the case of a 5 percent error in the control torques. That is, for $t > T = 260$ seconds the vehicle is being torqued at a different rate than are the estimator equations for angular velocities. To see the effects of the errors in the control torques, these figures may be compared to Figures 5.8 and 5.9 respectively.

In all of these examples it can be seen that the proposed control scheme does accomplish rate reduction. The results from these feasibility studies indicate that it may be possible to control the three angular velocities of a space vehicle using only one rate gyro.

5.11 Programs

Appendix J contains the listing of the programs which were used to produce the experimental results concerning the estimation and control of the angular velocities of a space

vehicle. While no claim is made concerning the efficiency, in terms of computation time, of these programs, they are working programs of some flexibility. A brief description of the important parameters of the programs is also contained in this appendix.

5.12 Conclusions and Future Work

A sequential least squares estimator has been formally derived. The approximations made were necessary in order to obtain the sequential estimator equations from the non-linear partial differential equation of invariant imbedding. This estimator could be implemented in real time.

In general, the question of observability of the system with respect to the output has been ignored, i.e., considering just the noisless case for the moment, does $y(t)$ for $0 \leq t \leq T$ uniquely define the state $x(T)$? The observability question for non-linear systems has received little attention in the technical literature. The sequential estimator presented in this chapter provides a tool for experimentally studying the observability of specific systems.

Examples were presented which demonstrate that the sequential least squares estimator proposed is feasible. The examples of specific interest to JPL concerned the

estimation of the three angular velocities about the principal axes of a rotating rigid body. These velocities were estimated based on noisy measurements on only one angular velocity. Examples were presented which demonstrate that rate reduction based on the estimated angular velocities, as produced by the sequential state estimator, is feasible.

Currently the following problems are being investigated: (1) The physical interpretation of the P equations in the state estimator, i.e., how should the initial conditions $P(0)$ be selected and precisely what interpretation should be given to $P(t)$. (2) The sensitivity question as it applies to the problem of interest to JPL, i.e., what happens to the performance of the estimator in the case when the principal moments of inertia are not known precisely, and can this error in the moments of inertia be compensated for?

CHAPTER 6

MIN-MAX OPTIMIZATION

6.1 Summary

In this chapter, the use of a min-max criterion in the solution of the specific optimal control problem is developed. The use of the criterion in conjunction with an approach analogous to differential approximation is discussed, and a method for solution of the problem using quasilinearization and linear programming is given. Linear programming and the simplex method are discussed briefly.

Several variations in approach to this problem are pointed out and discussed, and examples are given. Future work in this area is outlined.

6.2 Introduction

The method of differential approximation was developed in Chapter 3 of this report. The use of this method requires fitting one trajectory to another using a least-squares criterion. It is mentioned in Chapter 3 that an alternative to the use of the least-squares criterion for fitting the system

164

trajectory to the optimal trajectory is the use of a criterion which would minimize the maximum deviation between the two trajectories. Such a criterion shall be called a "min-max" criterion.

A modification of the differential approximation procedure as presented in Chapter 3 which makes use of the min-max criterion will be presented, along with a method for solving the problem which utilizes the basic ideas of quasilinearization [10], [Appendix B], and linear programming [33,34].

6.3 Use of Min-Max Criterion in SOC Problems

In Chapter 3 a typical optimal control problem was stated, and several of the classical methods for solving the problem were outlined. The limitations of these methods with regard to practical, realizable solutions were pointed out, and these limitations were in turn regarded as motivation for the SOC approach to the solution of the optimal control problem. Now, along with the approaches to the SOC problem given in Chapter 3, an additional approach will be presented.

Consider that the plant that one wishes to control is

described by the differential equation

$$\dot{\underline{x}} = \underline{f}(t, \underline{x}, u) \qquad (6.3.1)$$

where $\qquad \underline{x}$ - n-dimensional state vector

$\qquad\qquad\quad$ u - scalar control input

At the initial time (taken to be zero) the plant is specified

to be in a state $\underline{c}$, i.e.

$$\underline{x}(0) = \underline{c} \qquad (6.3.2)$$

It is desired to find the input $u = u(\underline{x}, t)$ which will

minimize the performance index

$$I = \int_0^T g(t, \underline{x}, u) \, dt \qquad (6.3.3)$$

In (6.3.3), g is a scalar, non-negative function, and T is

the terminal time, which may be fixed or free.

In order to convert this into an SOC problem, consider

a controller of the form

$$u = h(\underline{b}, \underline{x}) \qquad (6.3.4)$$

where h is a scalar function of known form and $\underline{b}$ is the m-

dimensional vector of parameters whose values constitute

166

the solution to the SOC problem. A restriction on the form

of h is that it be linear in the components of $\underline{b}$. The rea-

son for this restriction will become apparent subsequently.

In general h of equation (6.3.4) need not explicitly depend

on all the components of $\underline{x}$.

First it is necessary to find the optimum "open-loop"

solution, u*(t). If T is fixed, this can be done by solving

the canonic equations of the problem with appropriate boundary

conditions, usually by quasilinearization [10]. If T is

free, then some other method must be used, such as a gradient

method, as proposed in reference [6]. Having u*(t), one can

obtain $\underline{x}$*(t) the optimal trajectory, by integrating (6.3.1)

with initial conditions (6.3.2) and u*(t) as an input. How-

ever in general the latter step is unnecessary since any

method for determining u*(t) will also yield $\underline{x}$*(t).

Now the problem is to choose $\underline{b}$ so that the trajectory

$\underline{x}_s(t)$ obtained with the specific controller shall be fitted

to $\underline{x}$*(t) in the min-max sense. A modification of the quasi-

linearization scheme can be used to accomplish this task.

First, since $\underline{b}$ is a constant,

$$\dot{\underline{b}} = 0 \qquad\qquad (6.3.5)$$

Now one substitutes (6.3.4) into (6.3.1) and adjoins (6.3.5) to the resulting equation. One then has

$$\dot{\underline{x}} = \underline{f}(t, \underline{x}, h(\underline{b}, \underline{x}))$$

$$\dot{\underline{b}} = 0$$

(6.3.6)

with initial conditions $\underline{x}(0) = \underline{C}$ and no given conditions on $\underline{b}$. The object is to find a solution of (6.3.6) subject to (6.3.2) such that the value obtained for $\underline{b}$ causes $\underline{x}(t)$ to be fitted to $\underline{x}^*(t)$ in the min-max sense.

As in the quasilinearization solution to boundary-value problems, one can find a solution to this problem by forming a sequence of linear problems, the solutions to which will converge to the solution of (6.3.6) subject to the specified initial conditions.

Rewrite (6.3.6) as follows:

$$\dot{\underline{y}} = \underline{f}_1(t, \underline{y})$$

(6.3.7)

where

$$\underline{y} = \begin{bmatrix} \underline{x} \\ \underline{b} \end{bmatrix}$$

$$\underline{f}_1 = \begin{bmatrix} \underline{f} \\ \underline{0} \end{bmatrix}$$

$\underline{y}$ and $\underline{f}_1$ are $(n+m)$-dimensional vectors

168

Proceeding as in quasilinearization, one forms the "quasi-

linear" equations [Appendix B]

$$\underline{Y}_{k+1} = \underline{f}_1(t, \underline{Y}_k) + \frac{\partial f_1}{\partial \underline{y}} \bigg|_{\underline{Y}=\underline{Y}_k} (\underline{Y}_{k+1} - \underline{Y}_k) \qquad (6.3.8)$$

Some of the boundary conditions on (6.3.8) are

$$(y_1(0), y_2(0), \ldots, y_n(0))_{k+1} = (C_1, C_2, \ldots, C_n)$$

$$(6.3.9)$$

If the k-th approximation to the solution of (6.3.7),

$\underline{Y}_k$ is known a solution to (6.3.8) may be obtained in the

form

$$\underline{Y}_{k+1}(t) = \Phi_{k+1}(t) K_{k+1} + \underline{P}_{k+1}(t) \qquad (6.3.10)$$

where $\Phi_{k+1}(t)$ = fundamental matrix solution of the homogeneous part of equation (6.3.8) made unique by choosing $\Phi_{n+1}(0)$ = identity matrix

K_{k+1} = vector of initial conditions on $\underline{Y}_{k+1}$

$\underline{P}_{k+1}(t)$ = a particular solution of the inhomogeneous equation (6.3.8) made unique by choosing $\underline{P}_{k+1}(0) = \underline{0}$

At this point one departs from the usual quasilineariza-

tion procedure, which would involve constructing and solving

a system of linear algebraic equations in the components of
$\underline{K}$. Instead, one constructs a linear programming [33,34]
problem and solves it. The solution satisfies the min-max
criterion and gives the appropriate values of $\underline{b}$. This
process is iterated until convergence is obtained.

6.4 Linear Programming [33,34]

It is advantageous at this point to digress from the
problem at hand in order to give a brief explanation of
linear programming. The reader who is already familiar with
linear programming can skip this section and proceed directly
to section 6.5.

Consider the following problem. One has a system in
which there are n variables, x_1, x_2, ..., x_n. The proper-
ties of the system are such that the n variables are related
by m linear relations, which might be equalities or inequal-
ities. Also associated with the system is a quantity z
which represents some desired goal or objective; z should
be expressed as a linear combination of the variables x_i,
$i = 1, 2, ..., n$. It is desired that z be minimized (or
maximized) by a proper choice of the variables x_i.

As an illustration, suppose that the system under con-
sideration is a clothing store, and that the x_i represent

the amounts of various articles of clothing that the store manager will stock for sale in the store. Let z represent the total net profit, and assume that various linear relations are known (either empirically or otherwise) between the amounts of the various items purchased and the amounts that can be stored, the number of items that will be lost to shoplifters, damage in transit, etc., and the number of each item that can be expected to be sold. The problem here would be to pick the x_i such that all the relations cited above (constraints) would be satisfied and z, the profit, would be maximized.

In more precise mathematical terms, the problem could be stated in the following way. Given the objective form

$$z = \sum_{i=1}^{n} a_i x_i \qquad i = 1, 2, \ldots, n \qquad (6.4.1)$$

and the constraints

$$\sum_{i=1}^{n} b_{ij} x_i \leq c_j \qquad j = 1, 2, \ldots, m \qquad (6.4.2)$$

where a_i, b_{ij}, and c_j are constants for all i and j, find the values of the x_i which maximize (6.4.1) subject to the constraints (6.4.2).

The foregoing is a statement of the general linear programming problem. Of several methods available for solving the problem, the most popular and most generally used is the simplex method. In order to apply this method, certain restrictions must be applied to the problem, viz., the variables x_i must be non-negative, the constraints must be linear equalities, and the objective form must be minimized by the optimum solution. This constitutes the "standard form" of the linear programming problem.

It is an easy matter to transform the general problem of (6.4.1) and (6.4.2) into the more restricted form mentioned above. In order to insure the non-negativity of the variables, one takes advantage of the fact that any number can be written as the difference of two non-negative numbers. For instance, if the variable x_i in the original formulation of a problem has no restrictions on its sign, one makes the substitution

$$x_i = x_i' - x_i'' \qquad\qquad (6.4.3)$$

where $\qquad x_i' \geq 0 \quad , \quad x_i'' \geq 0$

A similar substitution is made for all variables whose non-negativity is not assured. Note that every such substitution

increases the number of variables in the linear programming problem by one.

If the original formulation of the problem contains inequality constraints, these may be converted to equality constraints by the introduction of non-negative "slack variables."

For example, suppose that the following constraints arise in a linear programming problem:

$$b_{11}x_1 + b_{12}x_2 \leq c_1$$
$$b_{21}x_1 + b_{22}x_2 \geq c_2$$

(6.4.4)

To convert (6.4.4) to equalities, the non-negative variables x_3 and x_4 would be introduced as follows:

$$b_{11}x_1 + b_{12}x_2 + x_3 = c_1$$
$$b_{21}x_1 + b_{22}x_2 - x_4 = c_2$$

(6.4.5)

Thus the inequalities become equalities. The variables x_3 and x_4 are called "slack variables" because they "take up the slack" in the inequalities. Again note that each slack variable introduced increases the number of variables to be considered.

The initial formulation of the linear programming problem may be such that the objective form is to be maximized. If such is the case then a change is necessary in order that the problem be in standard form. To make this change, one minimizes the negative of the original objective form, which is to say that maximizing z is equivalent to minimizing -z.

Thus it is seen that any linear programming problem can be put in standard form. The emphasis is placed on the standard form because it is necessary that a linear programming problem be in this form before the simplex method can be applied. A concise statement of the problem in standard form will now be given: Find the values of non-negative variables x_i which minimize the value of a linear form in the variables, subject to given linear equality constraints.

When the problem is put in standard form, the simplex method can be utilized to obtain the solution. An excellent presentation of the simplex method is given in Reference [33].

6.5 The Linear Programming Problem

To illustrate the formation of the linear programming problem mentioned previously, consider the following simplified situation. Suppose it is desired to fit $x_{1s}(t)$, the

first component of y_{k+1} of (6.3.10) to $x_1^*(t)$, the corresponding component of $\underline{x}^*(t)$, in the min-max sense. From (6.3.10), dropping the subscript k+1 for the sake of simplicity, one can write

$$x_{1s}(t) = \sum_{j=1}^{n+m} \varphi_{1,j} k_j + P_1 \qquad (6.5.1)$$

where $\qquad \varphi_{1,j}$ = j-th element of first row of Φ

$\qquad k_j$ = j-th component of $\underline{K}$

$\qquad P_1$ = first component of $\underline{P}$

A numerical solution to this problem requires that $x_{1s}(t)$ be fitted to a finite number of points on $x_1^*(t)$. Suppose that one has ℓ points on x_1^*, corresponding to ℓ different instants of time, i.e.,

$$x_1^*(t_i) = d_i \qquad i = 1, 2, \ldots, \ell \qquad (6.5.2)$$

From (6.5.1) one can write

$$x_{1s}(t_i) = \sum_{j=1}^{n+m} \varphi_{1,j}(t_i) k_j + P_1(t_i) \qquad (6.5.3)$$

The deviation between the two curves at each time t_i is $x_1^*(t_i) - x_{1s}(t_i)$. Let variables α_i and β_i be introduced

such that

$$\alpha_i \geq 0 \quad , \quad \beta_i \geq 0, \quad i = 1, 2, \ldots, \ell \qquad (6.5.4)$$

Now one may write the deviations as

$$x_1^*(t_i) - x_{1s}(t_i) = \alpha_i - \beta_i$$

or
$$(6.5.5)$$

$$d_i - \sum_{j=1}^{n+m} \varphi_{1,j}(t_i) \, k_j - P_1(t_i) = \alpha_i - \beta_i$$

$$i = 1, 2, \ldots, \ell$$

Note that the first n components of $\underline{K}$ are the initial conditions on $\underline{x}$, which are known; the remaining m components are the components of $\underline{b}$, which are to be found. Since the deviation between x_1^* and x_{1s} may be either positive or negative, one writes it as the difference of two non-negative variables, as in (6.5.5); the α_i and β_i will be variables in the linear programming problem and are required to be non-negative.

As the next step, introduce non-negative variables z, p_i, and q_i such that

$$\alpha_i \leq z$$

$$\beta_i \leq z$$

$$\alpha_i + p_i = z \tag{6.5.6}$$

$$\beta_i + q_i = z$$

$$i = 1, 2, \ldots, \ell.$$

Also, since non-negative variables are required, and generally one has no a priori knowledge of the signs of the components of $\underline{b}$, which are the last m components of $\underline{K}$, it is necessary to make the following substitution

$$k_j = k_j' - k_j'' \tag{6.5.7}$$

$$j = n+1, n+2, \ldots, n+m$$

From (6.5.7), (6.5.5), and (6.5.6), one has 3ℓ equations in the $(4\ell + 2m + 1)$ variables k_j', k_j'', α_i, β_i, p_i, q_i, and z $(j = n+1, n+2, \ldots, n+m$; $i = 1, 2, \ldots, \ell)$. Let z be eliminated from all but one of the equations, by standard pivoting operations [33].

One now has $(3\ell-1)$ equations in $(4\ell+2m)$ variables and z expressed as a linear combination of the variables. Notice that the equations are necessarily linear because of the

quasilinearization-type approach to the problem. This fits the standard form of the linear programming problem with z being the objective form that is to be minimized. Minimization of z will cause the maximum deviation between x_1^* and x_{1s} to be minimized.

The preceding formulation is somewhat cumbersome, as it involves more variables and equations than are actually necessary to solve the problem. Also, it is difficult to prove rigorously that in general, the maximum deviation is minimized when z is minimized, even though the examples show that this does occur. An alternate formulation is given, in which fewer variables and equations are involved, and the satisfaction of the min-max criterion by the solution is obvious.

Let the magnitude of the deviation between x_1^* and x_{1s} at time t_i be δ_i, i.e.

$$\left| x_1^*(t_i) - x_{1s}(t_i) \right| = \delta_i$$

(6.5.8)

$$i = 1, 2, \ldots, \ell$$

Relation (6.5.8) can be replaced by the following two relations:

178

$$x_1^*(t_i) - x_{1s}(t_i) \leq \delta_i$$

$$x_1^*(t_i) - x_{1s}(t_i) \geq -\delta_i \qquad (6.5.9)$$

$$i = 1, 2, \ldots, \ell$$

Now introduce the non-negative variable z such that

$$z \geq \delta_i \qquad , \qquad i = 1, 2, \ldots, \ell \qquad (6.5.10)$$

Using (6.5.10), one can eliminate the δ_i from (6.5.9), obtaining

$$x_1^*(t_i) - x_{1s}(t_i) \leq z$$

$$x_1^*(t_i) - x_{1s}(t_i) \geq -z \qquad (6.5.11)$$

$$i = 1, 2, \ldots, \ell$$

As in the previous formulation, it is necessary to make the substitution given by (6.5.7), since one generally has no a priori knowledge of the signs of the components of $\underline{b}$. Now let the non-negative variables p_i and q_i be introduced such that

$$x_1^*(t_i) - x_{1s}(t_i) + p_i = z$$

$$x_1^*(t_i) - x_{1s}(t_i) - q_i = -z \qquad (6.5.12)$$

$$i = 1, 2, \ldots, \ell$$

If z is eliminated from all but one of the equations (6.5.12), the result is $(2\ell-1)$ equations in the $(2\ell+2m)$ variables p_i, q_i, k_j', k_j'' ($i = 1, 2, \ldots, \ell$; $j = n+1, n+2, \ldots, n+m$), with z expressed as a linear combination of the variables. This is now a linear programming problem with z being the objective form that is to be minimized. From (6.5.9) and (6.5.10), the minimization of z will cause the maximum deviation to be minimized. Note that this formulation involves fewer equations and fewer variables than the first formulation.

It is not difficult to see that the size of the linear programming problem can get quite large in a solution of the type just described. For example, if one wanted to obtain values for two parameters by obtaining a min-max fit to twenty points, the linear programming problem would involve thirty-nine equations in forty-four variables using the second formulation. As the number of observations or parameters increases, the size of the linear programming problem increases. It is obvious that the amount of available storage in the computer places an upper limit on the size of the problem that can be considered.

One should be aware that the example cited above for purposes of illustration of the method of solution is greatly

simplified. For this procedure to be analogous to the differential approximation procedure of Chapter 3, it would be necessary to fit simultaneously all components of the state vector $(x_{1s}, x_{2s}, \ldots, x_{ns})$ to the corresponding components of the optimum solution $\underline{x}^*(t)$ in the min-max sense.

6.6 Examples

Examples 6.1:

The technique outlined above was used to solve a specific optimal control problem. The plant considered was described by the differential equations

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = -3x_2 - 2x_1 + u .$$

$$(6.6.1)$$

The initial conditions were

$$x_1(0) = -5.0$$

$$x_2(0) = -5.0$$

The performance index which was to be minimized was

$$\int_0^1 (x_1{}^2 + x_2{}^2 + u^2) \, dt \qquad (6.6.2)$$

The solution of (6.6.1) which minimizes (6.6.2) with the given initial conditions was found using quasilinearization.

The points to be used for the min-max fit were arbitrarily chosen at 0.1-second intervals on x_1^*, beginning at $t = 0.1$ sec. For this particular problem, five points were used. The controller used was of the form $u = bx_1$.

Convergence was obtained in three iterations, and the value of b was found to be -0.0247079.

The results are graphically shown in Figures 6.1 and 6.2. In Figure 6.1, $x_{1s}(t)$ and $x_1^*(t)$ are plotted. It can be seen that the two curves are quite close over the range used for the min-max fit and are somewhat divergent over the remainder of the time considered. The situation is similar for the derivative, x_2, which is shown in Figure 6.2.

Obviously, better agreement between x_{1s} and x_1^* could be obtained if more points were used for the min-max fit. This is done in the next example.

Example 6.2

The problem is the same as was considered in the previous example, except that ten points on x_1^* were used for the min-max fit. The points were taken at 0.1-second intervals, beginning at $t = 0.1$ sec. The value of b obtained

FIGURE 6.1

X₂ VS. TIME FOR EXAMPLE 6.1
——— OPTIMUM X₂
- - - - X₂ USING MIN—MAX FIT
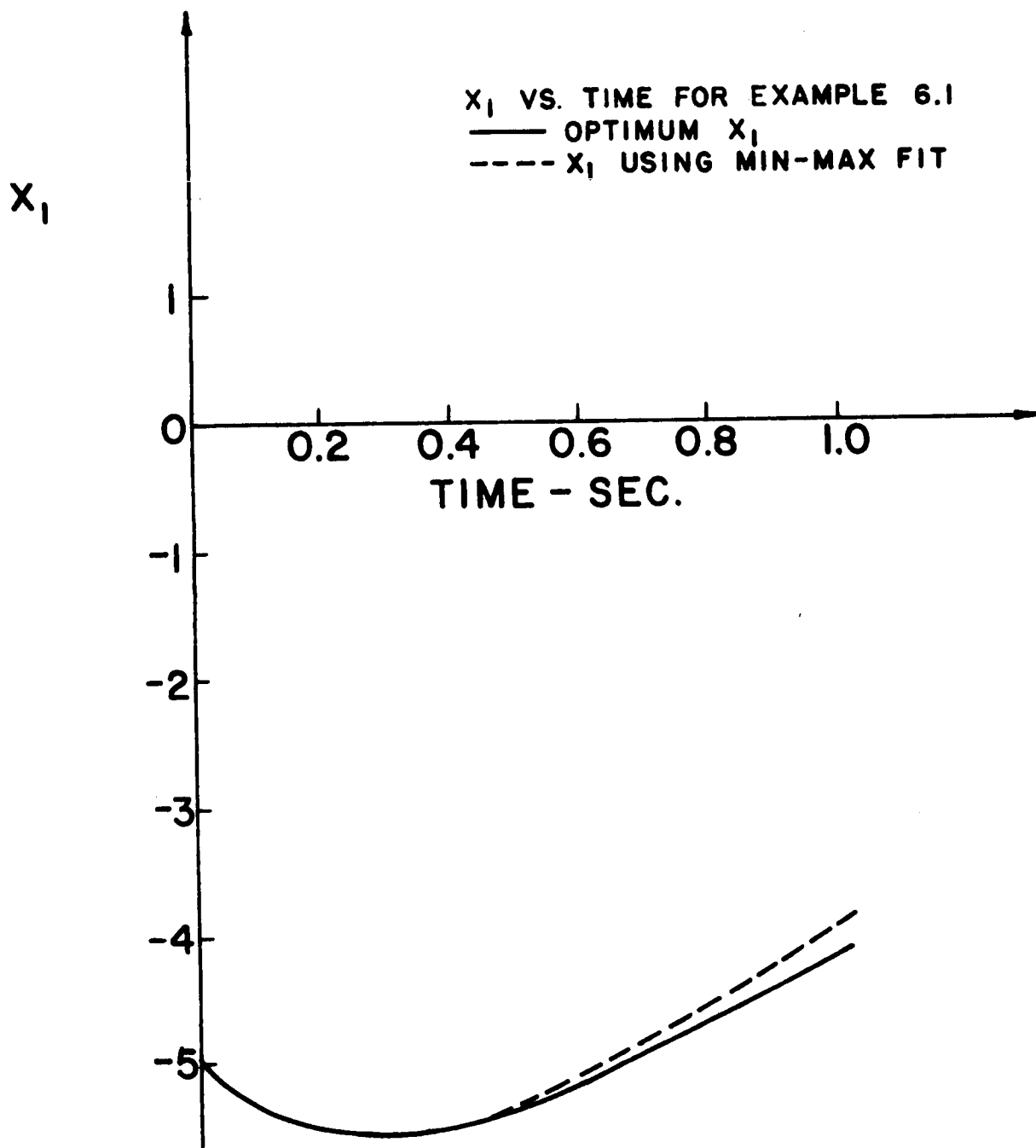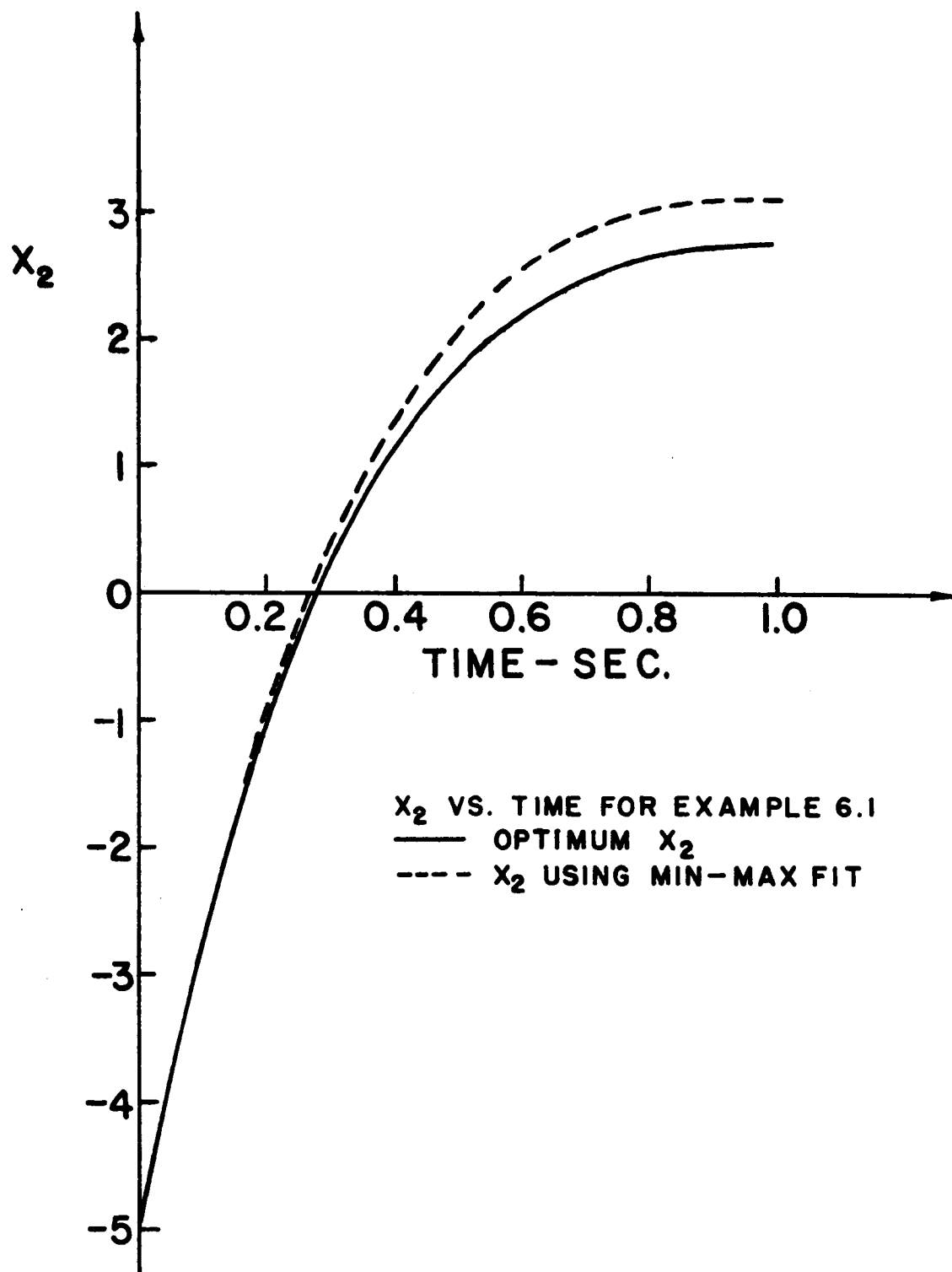
FIGURE 6.2

was -0.247892. Convergence occurred in three iterations.

Results are shown in Figures 6.3 and 6.4. As was expected, the agreement between x_{1s} and x_1* was much better than in the previous example; this is shown in Figure 6.3. Similarly, the derivative x_{2s} is in much better agreement with x_2* than in the previous example, as may be seen in Figure 6.4. The program used for example 6.2 is given in Appendix K.

In both the examples given above, better agreement between the specific trajectory (x_{1s}, x_{2s}) and the optimum trajectory (x_1*, x_2*) would have been obtained if a min-max fit had been performed simultaneously on both x_1 and x_2. As has been stated previously, this procedure would be necessary to make the min-max procedure analogous to the differential approximation procedure of Chapter 3 of this report.

6.7 Conclusions and Future Work

Several questions concerning the use of the min-max criterion in the solution of the SOC problem bear investigating. For instance, can satisfactory solutions to the SOC problem be obtained by performing min-max fits on selected components of the state vector rather than on all components

FIGURE 6.3

X₂ VS. TIME FOR EXAMPLE 6.2
—— OPTIMUM X₂
---- X USING MIN-MAX FIT
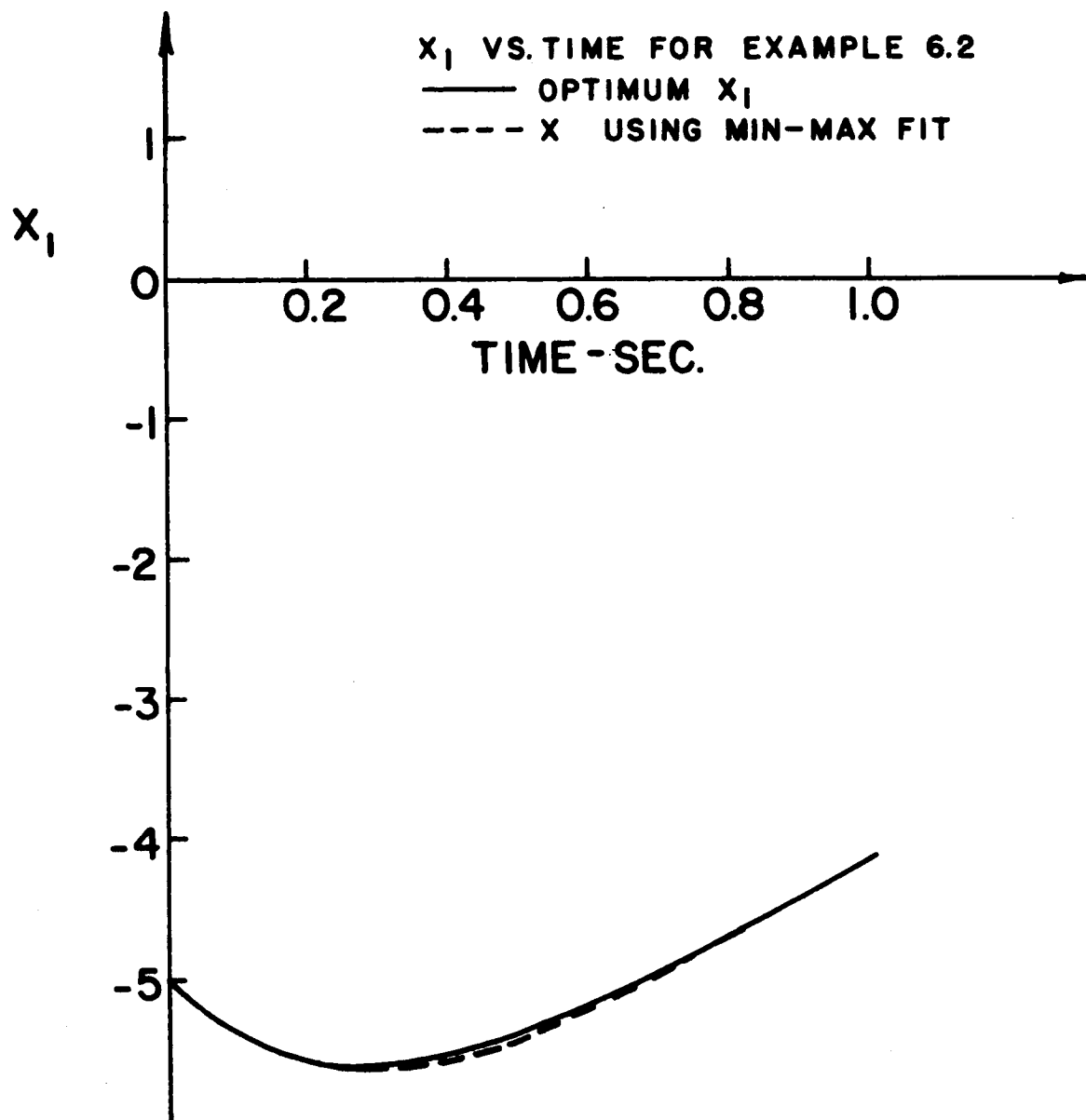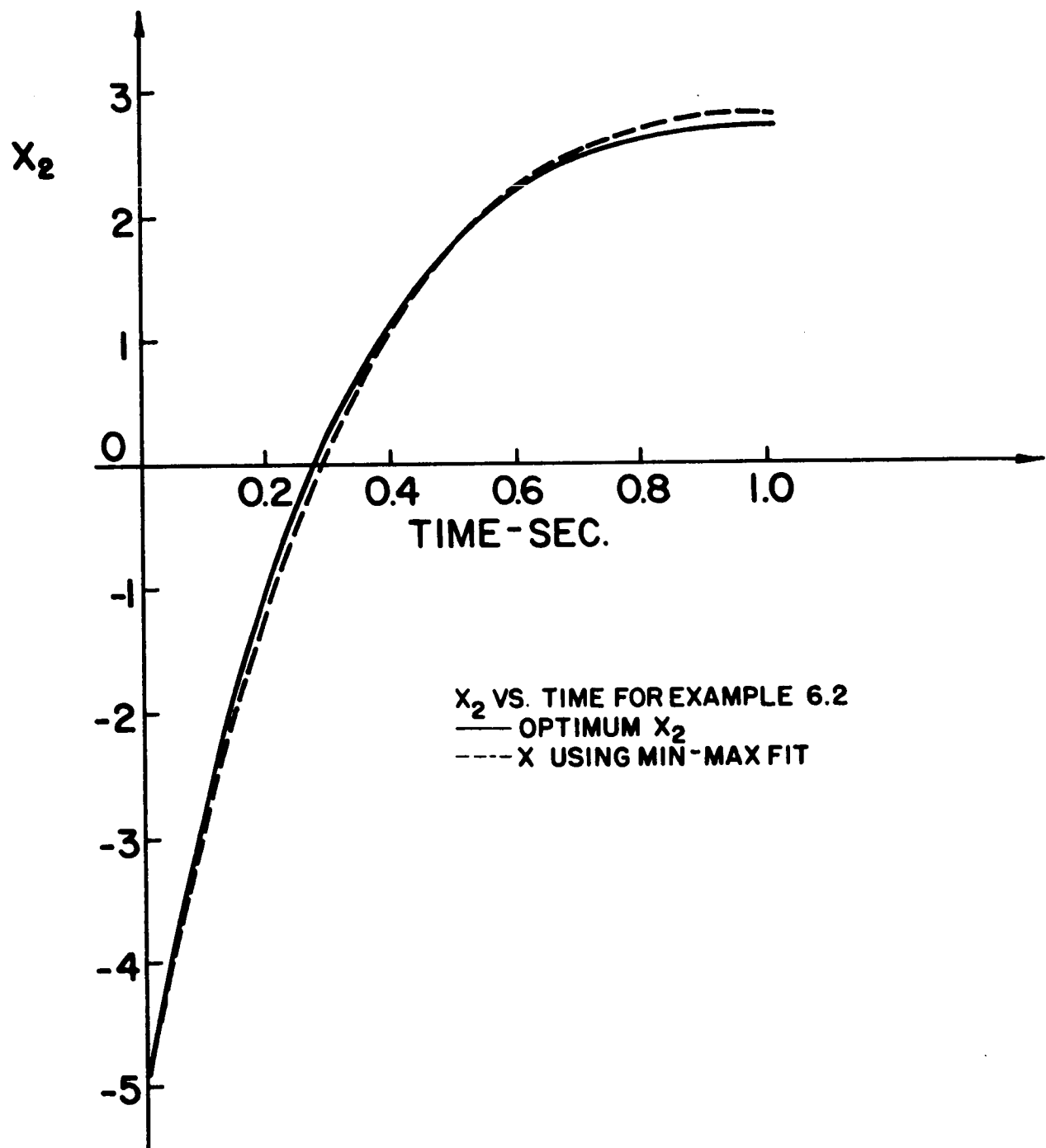
FIGURE 6.4

of the state vector? If so, which components should be selected? The examples given in the preceding section show that for the particular system considered, performing the min-max fit on one component provided a reasonably good fit of the second component. Another question which may be asked is the following. Rather than fit the components of the state vector to the respective components of the optimum state vector, can satisfactory and meaningful results be obtained by fitting the assumed form of the input, $h(\underline{b}, \underline{x})$ to the optimum control function $u*(t)$?

It is intended that the answers to these questions be sought through a series of experiments conducted on the digital computer. Also, it is deemed very important to discover ways in which the approach to the SOC problem developed in this chapter complements the other approaches mentioned elsewhere in this report. The min-max approach is certainly not considered to be a replacement for the other approaches.

CHAPTER 7

CONCLUSIONS

The material presented in this report is based on investigations which appear to hold considerable promise as far as the solution of the acquisition problem of a space vehicle is concerned. Much futher research is warranted before any strong claims can be made about the efficacy of using any of the techniques outlined here for actual design of the necessary controllers.

The reader will notice a certain amount of "disjointness" in the presentation in this report. This is essentially due to the many-pronged attack being made at Purdue in the effort of attempting to use optimal control theory to design practical and meaningful feedback controllers for the space vehicle. No doubt some of the techniques will prove to be barren while, hopefully, some will bear fruit.

The basic philosophy of the investigations is "digital experimentation" i. e. using the digital computer to run controlled experiments (principle of feedback in experimentation) in order to be able to deduce some properties of the structure of the solutions that one can expect in trying to solve the

nonclassical types of problems encountered in controlling

systems in a near optimal fashion based on partial information.

REFERENCES

1) Goldstein, H., <u>Classical</u> <u>Mechanics</u>, Chap. 5. Addison-Wesley Publishing Co., Inc., Reading, Mass, 1950.

2) Eckman, P. K., JPL Guidance and Control Technical Memo No. 4, Sec. 24, Feb. 10, 1960.

3) Bellman, R., R. Kalaba, and R. Sridhar, "Adaptive Control via Quasilinearization and Differential Approximation," The RAND Corporation, Research Memorandum RM-3928-PR, Nov. 1963.

4) Gleason, A., "A Search Problem on the n-cube," Proc. Symposia in Appl. Math., vol. 10, 1960.

5) Bellman, R., <u>Dynamic</u> <u>Programming</u>, Princetion University Press, Prinction, N. J., 1957.

6) Kelly, H. J., "Method of Gradients," in G. Leitmann (ed.), <u>Optimization</u> <u>Techniques</u> <u>with</u> <u>Applications</u> <u>to</u> <u>Aerospace</u> <u>Systems</u>, ch. 6, Academic Press, New York, N. Y., 1962.

7) Wilde, D. J., <u>Optimum</u> <u>Seeking</u> <u>Methods</u>, Prentice-Hall, Inc., Englewood Cliffs, N. J., 1964

8) Feldbaum, A. A., "Automatic Optimalizer," Automation and Remote Control, vol. 19, no. 8, 1958.

9) Bellman, R., H. Kagiwada, and R. Kalaba, "A Computational Procedure for Optimal System Design and Utilization," Proc. Nat. Acad. Sci., U. S. A., vol. 48, pp. 1524-1528, 1962.

10) Kalaba, R., "On Nonlinear Differential Equations, The Maximum Operation, and Monotone Convergence," J. Math. and Mech., vol. 8, pp. 519-574, 1959.

11) Fox, L., <u>Numerical</u> <u>Solution</u> <u>of</u> <u>Ordinary</u> <u>and</u> <u>Partial</u> <u>Differential</u> <u>Equations</u>, Addison-Wesley Publishing Co., Inc., Reading, Mass., 1962.

12) Bellman, R., <u>Perturbation</u> <u>Techniques</u> <u>in</u> <u>Mathematics</u>; <u>Physics</u>, <u>and</u> <u>Engineering</u>, Holt, Rinehart and Winston, New York, N. Y. 1964.

13) Bellman, R., R. Kalaba, and G. M. Wing, "Invariant Imbedding and the Reduction of Two-Point Boundary Value Problems to Initial Value Problems," Proc. Nat. Acad. Sci., U. S. A. vol. 46, 1960.

14) Bellman, R., H. Kagiwada, and R. Kalaba, "Numerical Studies of a Two-Point Nonlinear Boundary Value Problem Using Dynamic Programming, Invariant Imbedding, and Quasilinearization," The RAND Corporation, Reasearch Memorandum RM-4069-PR, March, 1964.

15) Bellman, R., and S. E. Dreyfus, Applied Dynamic Programming, Princeton University Press, Princton, N. J., 1962.

16) Courant, R., and D. Hilbert, Methods of Mathematical Physics, vol. I, Interscience Publishers, Inc., New York, N. Y., 1953.

17) Bliss, G. A., Lectures on the Calculus of Variations, The University of Chicago Press, Chicago, Ill., 1946.

18) Bellman, R., Adaptive Control Processes: A Guided Tour, Princeton University Press, Princetion, N. J., 1961.

19) Busnaw, D. W., "Differential Equations with Discontinous Forcing Term," Report 469, Experimental Towing Tank, Stevens Institute of Technology, Hoboken, N. J., Jan. 1953.

20) Kalman, R. E., and R. W. Koepcke, "Optimum Synthesis of Linear Sampling Control Systems Using Generalized Performance Indexes," Trans. ASME, vol. 80, pp. 1820-1826,1958.

21) Berkovitz, L. D., "Variational Methods in Problems of Control and Programming," J. Math. Anal. and Appl., vol. 3, no. 1, Aug. 1961, pp. 145-169.

22) Bellman, R.,"Successive Approximations and Computer Storage Problems in Ordinary Differential Equations," Comm. ACM, vol. 4, no. 5, May 1961, pp. 222-223.

23) Bellman, R., R. Kalaba, and B. Kotkin, "Differential Approximation Applied to the Solution of Convolution Equation," The RAND Corporation, Research Memorandum RM-3601-NIh, May 1963.

24) Pontryagin, L. S., V. G. Boltyanskii, R. V. Gamkrelidze, and E. F. Mishchenko, The Mathematical Theory of Optimal Processes, Interscience Publishers, New York, N. Y., 1962.

25) Bryson, A. E., S. E. Dreyfus, and W. E. Denham, "Optimal Programming Problem with Inequality Constraints: Necessary Conditions for Extremal Solutions," AIAA Journal, Nov. 1963.

26) Beckwith. R. E., "Analytic and Computational Aspects of Dynamic Programming Processes of High Dimension," Ph. D, Thesis, Purdue University, Lafayette, Indiana, 1959.

27) Unger, P., and H. Rubin, "Motion Under a Strong Restraining Force," Comm. in Pure and Appl. Math., New York University, 1955.

28) Katz, S. "Best Operating Points for Staged Systems," Industrial Engineering and Chemistry Fundamentals, vol. 1, no. 4, Nov. 1962.

29) Bellman, R., R. Kalaba, and R. Sridhar, "Sensitivity Analysis and Invariant Imbedding," The RAND Corporation, Research Memorandum RM-4039-PR, March 1964.

30) Bellman, R., H. Kagiwada, R. Kalaba, and R. Srldhar, Invariant Imbedding and Non-Linear Filtering Theory," The RAND Corporation, Research Memorandum RM-4374-Pr, Dec. 1964.

31) Bryson, A. E., and Frazier, M., "Smoothing for Linear and Nonlinear Dynamic Systems," Proc. Of the Optimum Systems Synthesis Conference, Wright-Patterson AFB, Ohio, AST-TDR-63-119, Sept. 1962.

32) Cox, H., "Estimation of State Variables Via Dynamic Programming," Pre-Prints of Technical Papers, Fifth JACC, June, 1964, pp. 376-381.

33) Dantzig, G. B., Linear Programming and Extensions, Princeton University press, Princeton, N. J., 1963.

34) Kalaba, R., "Linear Programming," The RAND Corporation, Paper P-950, Oct. 1, 1956.

APPENDIX A

OPTIMAL CONTROL SYSTEMS

A.1 Introduction

The analysis and design of control systems have relied heavily in the past on empirical methods such as Nyquist diagrams, Bode plots, Root loci, etc. These employ linearized models and provide the designer with qualitative information regarding the effect of the controller (equalizer) on the response of the system.

More recently, attempts have been made in the automatic control literature to develop analytical methods for analysis and design of control systems. Several of these attempts have focused attention on the possibility of using methods from the calculus of variations in designing control systems.

Briefly, optimal control theory is concerned with a variational formulation of automatic control problems and the attempt to solve the resulting problem using methods from the calculus of variations. Undoubtedly the number of significant engineering control problems solved to date using optimal control theory is small. However, the methods hold so much promise that they seem now to warrant serious study on the part of practicing control engineers.

194

In this presentation, no special attention will be paid

to mathematical rigor since rigorous justification of each

and every step in the derivations will obscure the main

philosophy of the methods.* The presentations will rely on

the method of dynamic programming developed by Bellman rather

than the more usual calculus of variations.** This approach

permits a greater simplicity in derivations with a slight

sacrifice in rigor.

A Typical Optimal Control Problem

A typical optimal control problem is the following:

The plant or object to be controlled is described by a vector

differential equation of the form

$$\dot{\underline{x}} = \underline{f}(t, \underline{x}, \underline{u}) \qquad \qquad (A.1.1)$$

where $\underline{x}$ is a n-vector, $(x_1, \ldots, x_n)$ called the state vector

of the plant and the components x_i, $i = 1, 2, \ldots, n$ are

called the state variables. $\underline{x}(t)$ is the state of the plant

at time t. $\underline{u}$ is a m-vector $(u_1, \ldots, u_m)$ and is called the

---

*The spirit of the derivations and discussion in this presentation are along the lines of Chapter $\underline{V}$ of [15].

**The reader uninitiated in the methods of Calculus of Variations will find it worthwhile to refer to the excellent discussion in Chapter IV of [16]. For an advanced and rigorous discussion of the classical Calculus of Variations refer to [17]. For additional information on dynamic programming methods, refer to [5], [18].

<u>control</u> <u>vector</u> of the plant. The components $u_i(t)$, $i=1,2,\ldots,m$ are called control functions. $\underline{f}$ is a n-vector $(f_1, \ldots, f_n)$. The f_i, $i = 1, 2, \ldots, n$ are assumed to possess continuous first partial derivatives with respect to all their arguments.

The control functions may be either unconstrained or may be required to fall within an allowable range of values; typically $\underline{u}(t)$ may be required to satisfy the inequality

$$k_{1i} \leq u_i(t) \leq k_{2i}, \quad i=1,2\ldots,m \qquad (A.1.2)$$

The general constraint on u(t) (an example of which is (A.1.2)) will be symbolically denoted by $\underline{u} \in \Omega$ where Ω is a suitably defined set.

In most applications the $u_i(t)$ are required to be at least piecewise continuous thus excluding, for example, impulse control.

The situation is summarized by the statement that if the state $\underline{x}$ is given at some initial instant in the form

$$\underline{x}(t_o) = \underline{c}_o \qquad (A.1.3)$$

and the control $\underline{u}(\tau)$ given for $t_o \leq \tau \leq t$, then the state $\underline{x}(t)$ is determined uniquely. This assumption implies for example that Eq. (A.1.1) cannot be a delay-differential equation.

The present theory assumes that

1) the plant is completely controllable, a condition that is rather difficult to establish in the case of nonlinear time-varying plants;

2) all the state variables are available for measurement; and

3) that disturbances of any kind are negligible.

Such assumptions can seldom be justified in practice because either

a) some of the state variables are not accessible for measurement, or

b) some of the state variables that can be measured are contaminated with noise, or

c) both of the above reasons.

Nevertheless one can still utilize the results of optimal control theory by either building the best possible estimator in the case of noise-corrupted state variables or obtaining by linear filters (equalizers) the best possible approximate values of the state variables that cannot be measured; an example of this is shown in Fig. A.1.
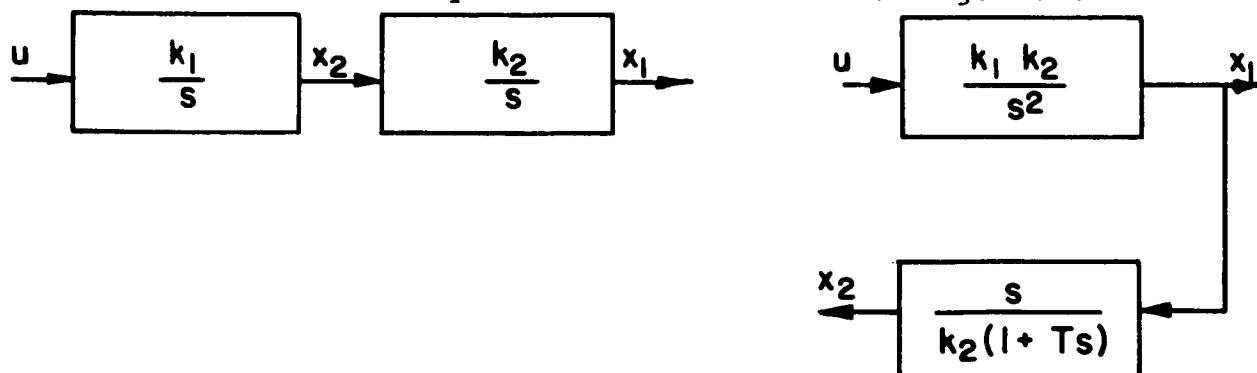
T as small as possible, $T > 0$

FIGURE A.1

The control problem is to find $\underline{u}(t)$ such that a given functional of $\underline{x}(t)$ and $\underline{u}(t)$ (the _index of performance_ or _return function_) is minimized. The present theory allows indices to take the form

$$I_1(\underline{u}) = \int_{t_o}^{T} g_1(t, \underline{x}(t), \underline{u}(t)) \, dt + h(T, \underline{x}(T)) \quad (A.1.4)$$

In equation (A.1.4) g_1 and h are scalar valued functions of their arguments and are assumed to possess continuous first partial derivatives with respect to all their arguments. The _terminal time_ T may be fixed or free; in general the terminal state $\underline{x}(T)$ will be restricted to a given region of the state space.

The right hand side of equation (A.1.4) consists of two parts which may be intuitively interpreted as consisting of (1) a part which allows for costs which accumulate over the entire duration of the process and (2) a part which allows for costs incurred due to deviations from desired states when the process terminates.

A control $\underline{u}^*(t)$, $t_o \leq t \leq T^*$ for which $I_1(u)$ is a minimum (subject to (A.1.1), (A.1.2) and initial and terminal conditions on the state of the system) is called optimal; the corresponding state trajectory $\underline{x}^*(t)$, $t_o \leq t \leq T^*$ is

called an optimal trajectory. Here T* = T if the terminal time is fixed and T* is equal to that value of T which minimizes (A.1.4).

In most practical applications the choice of the actual performance index is not obvious. Considerable physical insight into the actual process to be controlled is necessary before a meaningful and acceptable performance index can be determined. The mathematical theory does not aid in picking a suitable performance index. This is where engineering experience comes to the fore. The minimization of a certain performance index may yield a satisfactory system for a particular plant to be controlled, whereas the same performance index applied to design an optimum controller for a different plant may yield a considerably different final system.

For the processes where meaningful performance indices cannot be obtained in a form suitable for application of optimal control theory based upon engineering experience, considerable research effort is currently reported on optimization of control systems using several performance indices. While the research may eventually provide the answer on how to choose the most meaningful performance index for a given

control application, at the present state of the art one still has to make an arbitrary choice of a performance index and use it as a design tool.

The performance index (A.1.4) can be rewritten in the form

$$I_1(\underline{u}) = \int_{t_O}^{T} \left[g_1(t, \underline{x}(t), \underline{u}(t)) + \frac{dh}{dt}(t, \underline{x}(t)) \right] dt$$

$$+ h(t_O, \underline{x}(t_O)) \qquad\qquad (A.1.5)$$

$$= I_2(\underline{u}) + h(t_O, \underline{x}(t_O)) \qquad (A.1.6)$$

where

$$I_2(\underline{u}) = \int_{t_O}^{T} g(t, \underline{x}(t), \underline{u}(t)) \, dt \qquad (A.1.7)$$

and

$$g(t, \underline{x}(t), \underline{u}(t)) = g_1(t, \underline{x}(t), \underline{u}(t)) + \frac{dh}{dt}(t, \underline{x}(t))$$

$$(A.1.8)$$

Since for the optimization problems to be considered here, the initial time t_O and initial state $\underline{x}(t_O)$ are fixed, $h(t_O, \underline{x}(t_O))$ is a constant in the right hand side of (A.1.6). Hence minimizing $I_1(\underline{u})$ is equivalent to minimizing $I_2(\underline{u})$, i.e., the same optimal control $\underline{u}^*(t)$ will result when minimizing either functional. Henceforth it will be assumed that the performance index to be minimized is of the form

(A.1.7).

In the following section, a basic partial differential equation associated with the optimization problem will be derived. This equation is called the functional equation of dynamic programming.

The type of problem considered in this section is often referred to in the literature as the _regulator problem_.

A.2 The Partial Differential Equation of Dynamic Programming

A basic partial differential equation associated with the optimization problem will be derived next. To do this, the method of invariant imbedding will be used. The underlying idea for invariant imbedding is the following. Faced with the problem of determining certain properties of one particular process, one may attempt to do so by considering that one process in isolation. However, it is often profitable and simpler to consider a whole family of processes of which the original process is a member and try to interconnect the properties of neighboring processes. This is invariant imbedding. Many structural properties of the given process can be determined using this technique.

The root locus method is an example of a technique which may be interpreted in terms of invariant imbedding. Often one is interested in determining the location of the closed loop

poles of a <u>particular fixed value</u> of open loop gain. This problem may be imbedded in a general class of problems in which the open loop gain may be a variable and the closed loop pole locations may now be desired. The solution to the general problem will yield the solution to the original problem. One method of solving the general problem is of course, the root locus method.

For example, instead of considering the specific optimization problem outlined in the previous section, consider the following more general problem. In this derivation it will be assumed that the terminal time T is fixed.

The plant is still described by equation (A.1.1). However the "initial" state of the plant at time τ is given by

$$x_i(\tau) = C_i \qquad i = 1, 2, \ldots, n \qquad \text{(A.2.1)}$$

where $-\infty < C_i < \infty$.

It is desired to choose $\underline{u}(t)$, $\tau \leq t \leq T$ such that a performance index of the form

$$I(\underline{u}) = \int_{\tau}^{T} g(t, \underline{x}, \underline{u}) \, dt \qquad \text{(A.2.2)}$$

is minimized.

In equation (A.2.2)

$$-\infty < \tau \leq T$$

Notice that the original problem has been "imbedded" in a general class of problems. The special case of this general class when $\tau = t_o$ and $\underline{C} = \underline{C}_o$ reduces to the original optimization problem.

Since the minimum value of the performance index depends on the initial state $\underline{C}$ and the starting instant τ, define the "<u>return function</u>" or "<u>value function</u>" $J(\underline{C}, \tau)$ as

$$J(\underline{C}, \tau) = \underset{\underline{u}(t) \, \epsilon \, \Omega}{\text{Min}} \int_{\tau}^{T} g(t, \underline{x}, \underline{u}) \, dt \qquad \text{(A.2.3)}$$

subject to the differential constraint (A.1.1) for a process starting at time τ with initial state $\underline{C}$. If constraints of the form (A.1.2) are specified, the $\underline{u}(t)$ which minimizes the integral is required to satisfy these constraints. This will be symbolically denoted by $\underline{u}(t) \, \epsilon \, \Omega$.

To proceed further, it is necessary to use the <u>principle of optimality</u> due to Bellman. It is stated as follows:

<u>The Principle of Optimality</u>. An optimal policy has the property that whatever the initial state and initial decisions are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision.

The principle of optimality may be viewed as a means for obtaining the optimal "<u>policy</u>" for a "<u>multi-stage decision process</u>."

Hence, to use the principle of optimality in the problem of interest here, the optimization problem should be reformulated as a multi-stage decision process. This can be done as follows.

Rewrite equation (A.2.3) as

$$J(\underline{C}, \tau) = \min_{\substack{\underline{u}(t) \in \Omega \\ \tau \leq t \leq T}} \left[\int_{\tau}^{\tau+\Delta} g(t, \underline{x}, \underline{u}) \, dt + \int_{\tau+\Delta}^{T} g(t, \underline{x}, \underline{u}) \, dt \right]$$

$$(A.2.4)$$

The choice of the control function (or decision) $\underline{u}(t)$ in the interval $\tau \leq t \leq T$ so as to minimize the quantity in the braces in the right hand side of (A.2.4) may be viewed as a choice of $\underline{u}(t)$ in the first stage $\tau \leq t > \tau + \Delta$ and the choice of $\underline{u}(t)$ over the remaining stages $\tau + \Delta < t \leq T$.

From the principle of optimality, whatever the choice of $\underline{u}(t)$ in the first stage, the decision $\underline{u}(t)$, $\tau + \Delta < t \leq T$ must be optimal with regard to the state resulting from the first decision.

Now for the arbitrary decision $\underline{u}(t)$, $\tau \leq t \leq \tau + \Delta$ for a system in state $\underline{x}(\tau) = \underline{C}$, the state at time $(\tau + \Delta)$ can be determined from equation (A.1.1). Denoting this state by $\underline{x}(\tau + \Delta)$, equation (A.1.1) yields

$$\underline{x}(\tau + \Delta) = \underline{C} + \underline{f}(\tau, \underline{C}, \underline{u}(\tau)) \cdot \Delta + 0(\Delta^2) \qquad (A.2.5)$$

for Δ sufficiently small.

204

In equation (A.2.5), $O(\Delta^2)$ consists of terms which have the property

$$\operatorname*{Lim}_{\Delta \to 0} \frac{O(\Delta^2)}{\Delta} = 0 \tag{A.2.6}$$

An optimal choice of $\underline{u}(t)$, $\tau + \Delta < t \leq T$ will yield

$$\operatorname*{Min}_{\substack{\underline{u}(t) \in \Omega \\ \tau + \Delta < t \leq T}} \int_{\tau+\Delta}^{T} g(t, \underline{x}, \underline{u}) dt = J(\underline{x}(\tau+\Delta), (\tau+\Delta)) \tag{A.2.7}$$

for a plant which at time $(\tau + \Delta)$ is in the state given by equation (A.2.5).

From equations (A.2.4) and (A.2.7) and the principle of optimality

$$J(\underline{C}, \tau) = \operatorname*{Min}_{\substack{\underline{u}(t) \in \Omega \\ \tau \leq t \leq \tau + \Delta}} \left[\int_{\tau}^{\tau+\Delta} g(t, \underline{x}, \underline{u}) dt + J(\underline{x}(\tau+\Delta), (\tau+\Delta)) \right] \tag{A.2.8}$$

Now

$$\int_{\tau}^{\tau+\Delta} g(t, \underline{x}, \underline{u}) dt = g(\tau, \underline{C}, \underline{u}(\tau)) \cdot \Delta + O(\Delta^2) \tag{A.2.9}$$

Also, using equation (A.2.5)

$$J(\underline{x}(\tau + \Delta), \tau + \Delta) = J(\underline{C} + \underline{f}(\tau, \underline{C}, \underline{u}(\tau)) \cdot \Delta + O(\Delta^2), \tau + \Delta)$$

$$\tag{A.2.10}$$

Expanding the right hand side of equation (A.2.10) about

$(\underline{C}, \tau)$ using Taylor's formula yields

$$J(\underline{x}(\tau + \Delta), \tau + \Delta) = J(\underline{C}, \tau) + < \underline{f}(\tau, \underline{C}, \underline{u}(\tau)), \nabla_{\underline{C}} J > \Delta$$
$$+ \frac{\partial J}{\partial \tau} \cdot \Delta + O(\Delta^2) \quad \text{(A.2.11)}$$

In equation (A.2.11), $< , >$ represents the Euclidean inner product of two vectors and $\nabla_{\underline{C}} J$ represents the gradient of J with respect to $\underline{C}$ defined as the n-dimensional vector

$$\left(\frac{\partial J}{\partial c_1}, \frac{\partial J}{\partial c_2}, \cdots, \frac{\partial J}{\partial c_n} \right).$$

Substituting from equations (A.2.9) and (A.2.11) in (A.2.8) yields

$$J(\underline{C}, \tau) = \underset{\substack{\underline{u}(t) \in \Omega \\ \tau \le t \le \tau + \Delta}}{\text{Min}} \left[g(\tau, \underline{C}, \underline{u}(\tau)) \cdot \Delta + J(\underline{C}, \tau) \right.$$
$$\left. + < \underline{f}(\tau, \underline{C}, \underline{u}(\tau), \nabla_{\underline{C}} J > \cdot \Delta + \frac{\partial J}{\partial \tau} \cdot \Delta + O(\Delta^2) \right]$$
$$\text{(A.2.12)}$$

Since $J(\underline{C}, \tau)$ in the right hand side of equation (A.2.12) is independent of $\underline{u}(t)$, it can be moved outside the minimization operation. Hence

$$\underset{\substack{\underline{u}(t) \in \Omega \\ \tau \le t \le \tau + \Delta}}{\text{Min}} \left[g(\tau, \underline{C}, \underline{u}(\tau)) \, \Delta + < \underline{f}(\tau, \underline{C}, \underline{u}(\tau)), \, \nabla_{\underline{C}} J > \Delta \right.$$

$$\left. + \frac{\partial J}{\partial \tau} \Delta + O(\Delta^2) \right] = 0$$

Dividing throughout by Δ and considering the limit when $\Delta \to 0$ yields

$$\frac{\partial J}{\partial \tau} + \underset{\underline{u}(\tau) \in \Omega}{\text{Min}} \left[g(\tau, \underline{C}, \underline{u}(\tau)) + < \underline{f}(\tau, \underline{C}, \underline{u}(\tau)), \, \nabla_{\underline{C}} J > \right] = 0$$

$$(A.2.13)$$

Replacing τ by t and $\underline{C}$ by $\underline{x}$ in equation (A.2.13), it can be rewritten as

$$\frac{\partial J}{\partial t} + \underset{\underline{u}(t) \in \Omega}{\text{Min}} \left[g(t, \underline{x}, \underline{u}(t)) + < \underline{f}(t, \underline{x}, \underline{u}(t)), \, \nabla_{\underline{x}} J > \quad = 0 \right]$$

$$(A.2.14)$$

The return function thus has to satisfy equation (A.2.14). Equation (A.2.14) is called the <u>functional equation of dynamic programming</u>. When the minimization is performed and the term within the square brackets replaced by its minimum value, the resulting equation is called the <u>Hamilton-Jacobi partial differential equation</u>.

Notice that by the definition of the return function

(A.2.3), the boundary condition on (A.2.14) is

$$J(\underline{x}, T) = 0 \qquad\qquad (A.2.15)$$

To conform with the accepted terminology in optimal control literature, the functional equation of dynamic programming will be written using the so-called <u>Hamiltonian</u> for the minimization problem.

The scalar valued function H(t, $\underline{x}$, $\underline{u}$, $\underline{\lambda}$) called the Hamiltonian is defined as

$$H(t, \underline{x}, \underline{u}, \underline{\lambda}) = g(t, \underline{x}, \underline{u}) + < \underline{\lambda}, \underline{f}(t, \underline{x}, \underline{u}) >$$
$$(A.2.16)$$

where $\underline{\lambda}$ is an arbitrary n-dimensional vector $(\lambda_1, \ldots, \lambda_n)$ called the <u>Lagrange multiplier vector</u>.

From (A.2.14) and (A.2.16)

$$\frac{\partial J}{\partial t} + \underset{\underline{u}(t) \, \epsilon \, \Omega}{\text{Min}} \; H(t, \underline{x}, \underline{u}, \nabla_{\underline{x}}J) = 0 \qquad (A.2.17)$$

Denote the value of $\underline{u} \, \epsilon \, \Omega$ which instantaneously minimizes the Hamiltonian, eq. (A.2.16) by $\underline{u}^*$. This minimization will yield $\underline{u}^*$ explicitly (at least in principle) in the form

$$\underline{u}^* = \underline{u}^* \ (t, \ \underline{x}, \ \underline{\lambda}) \qquad\qquad \text{(A.2.18)}$$

Define

$$H^*(t, \ \underline{x}, \ \underline{\lambda}) = H(t, \ \underline{x}, \ \underline{u}, \ \underline{\lambda}) \ \Big|_{\underline{u} \ = \ \underline{u}^* \ (t, \ \underline{x}, \ \underline{\lambda})}$$

$$\text{(A.2.19)}$$

Thus $H^*(t, \ \underline{x}, \ \underline{\lambda})$ is the minimum value of the Hamiltonian with respect to $\underline{u} \ \epsilon \ \Omega$. In terms of this minimum value of the Hamiltonian, equation (A.2.17) is equivalent to

$$\frac{\partial J}{\partial t} + H^*(t, \ \underline{x}, \ \nabla_{\underline{x}} J) = 0 \qquad\qquad \text{(A.2.20)}$$

Equation (A.2.20) is the <u>Hamilton-Jacobi partial different-</u><u>ial</u> equation for the optimization problem.

Equation (A.2.20) when solved with the boundary condition (A.2.15) will yield $J(\underline{x}, \ t)$ as a solution. Knowledge of $J(\underline{x}, \ t)$ implies that the value of the performance index is determined for a process starting at any time with any initial state.

From equations (A.2.16) and (A.2.17) it is evident that on an optimal trajectory, the Lagrange multiplier vector $\underline{\lambda}$ can be expressed in the form

$$\underline{\lambda} = \nabla_x J \qquad (A.2.21)$$

Substituting from (A.2.21) in (A.2.18) will result in $\underline{u}^*$ determined explicitly as a function of t, the current time and $\underline{x}$, the current state. This then will truly be a <u>feedback solution</u> to the optimization problem.

Hence, if the Hamilton-Jacobi equation can be explicitly solved, the optimal feedback solution can be obtained.

Example

Consider a linear time invariant plant governed by

$$\underline{\dot{x}} = A \underline{x} + \underline{b} u \qquad (A.2.22)$$

where A is a n x n matrix and $\underline{b}$ is a n-vector. In equation (A.2.22) u is a scalar, which implies that the plant has only one input. Assume that u is unconstrained.

Let the plant be in an initial state

$$\underline{x}(0) = \underline{c} \qquad (A.2.23)$$

It is required to find a feedback solution, i.e., u as a function of the current state and possibly current time so as to minimize a performance index of the form

210

$$I(u) = \frac{1}{2} \int_0^\infty \left[< \underline{x}, \ Q \underline{x} > + \alpha u^2 \right] dt \qquad (A.2.24)$$

In equation (A.2.24), Q is assumed to be a constant positive semi-definite matrix and α is a positive constant.

From equation (A.2.16), the Hamiltonian for this problem is

$$H(t, \underline{x}, u, \underline{\lambda}) = \frac{1}{2} < \underline{x}, \ Q \underline{x} > + \frac{\alpha}{2} u^2 + < \underline{\lambda}, \ A \underline{x} + \underline{b} u >$$

$$= \frac{1}{2} < \underline{x}, \ Q \underline{x} > + < \underline{\lambda}, \ A \underline{x} > + \frac{\alpha}{2} u^2 + < \underline{\lambda}, \ \underline{b} > u$$

$$(A.2.25)$$

To find H* of (A.2.19), the value u* which minimizes H has to be determined. Since u is unconstrained, this can be simply done by equating to zero the partial derivative of H with respect to u. This step yields, from equation (A.2.25)

$$\alpha u* + < \underline{\lambda}, \ \underline{b} > = 0$$

i.e.,

$$u* = - \frac{1}{\alpha} < \underline{\lambda}, \ \underline{b} > \qquad (A.2.26)$$

Hence, corresponding to (A.2.19)

$$H^*(t, \underline{x}, \underline{\lambda}) = \frac{1}{2} < \underline{x}, \ Q \ \underline{x} > + < \underline{\lambda}, \ A \ \underline{x} > - \frac{1}{2\alpha} \ (< \underline{\lambda}, \ \underline{b} >)^2$$

$$(A.2.27)$$

From the fact that the plant and the coefficients in the integrand of the performance index are time-invariant and the optimization is for an infinite duration process, it follows from the definition of the return function, equation (A.2.3) that $J(\underline{x}, t)$ will depend only on the initial state $\underline{x}$. This implies that

$$\frac{\partial J}{\partial t} = 0 \qquad\qquad (A.2.28)$$

From (A.2.20), (A.2.27) and (A.2.28), the Hamilton-Jacobi equation is

$$\frac{1}{2} < \underline{x}, \ Q \ \underline{x} > + < \nabla_{\underline{x}} J, \ A \ \underline{x} > - \frac{1}{2\alpha} \ (< \nabla_{\underline{x}} J, \ \underline{b} >)^2 = 0$$

$$(A.2.29)$$

Assume a solution to (A.2.29) of the form

$$J = < \underline{x}, \ P \ \underline{x} > \qquad\qquad (A.2.30)$$

where P is an unknown positive definite constant matrix. Then

212

$$\nabla_{\underline{x}} J = 2P \underline{x} \qquad\qquad (A.2.31)$$

Substituting from (A.2.31) in (A.2.29)

$$\frac{1}{2} < \underline{x}, \ Q \ \underline{x} > + \ 2 < P \ \underline{x}, \ A \ \underline{x} > - \frac{1}{2\alpha} \left(2 < P \ \underline{x}, \ \underline{b} >\right)^2 = 0$$

i.e.,

$$< \underline{x}, \ \frac{1}{2} Q \ \underline{x} > + < \underline{x}, \ P A \ \underline{x} > + < \underline{x}, \ A^T P \ \underline{x} > - \frac{2}{\alpha} < \underline{x}, \ P \ \underline{b} \ \underline{b}^T P \ \underline{x} > = 0$$

i.e.,

$$< \underline{x}, \ \left(\frac{1}{2} Q + PA + A^T P - \frac{2}{\alpha} P \ \underline{b} \ \underline{b}^T P\right) \underline{x} > = 0$$

This implies that the matrix P should satisfy the algebraic equation

$$\frac{1}{2} Q + PA + A^T P - \frac{2}{\alpha} P \ \underline{b} \ \underline{b}^T P = 0 \qquad (A.2.32)$$

Equation (A.2.32) is equivalent to $n(n + 1)/2$ simultaneous equations involving the $n(n + 1)/2$ unknown elements of the symmetric matrix P. The solution of equation (A.2.32) will determine the matrix P and consequently the return function (A.2.30).

Also $\nabla_{\underline{x}} J$ of equation (A.2.31) will be determined. From equation (A.2.21) it is seen that $\underline{\lambda}$ will also be determined. Hence the optimal feedback law, from equations (A.2.31), (A.2.26) and (A.2.21) is

$$u^* = -\frac{1}{\alpha} < 2P \underline{x}, \underline{b} >$$

$$= -\frac{2}{\alpha} < P \underline{b}, \underline{x} > \qquad (A.2.33)$$

From equation (A.2.33) it is seen that the optimal controller is a linear time invariant feedback controller which requires measurement of all the states of the plant.

For numerical evaluation consider specifically the double integration plant,

$$\dot{x}_1 = x_2 \qquad\qquad x_1(0) = c_1, \ x_2(0) = c_2$$

$$\dot{x}_2 = u \qquad\qquad x_1(\infty) = x_2(\infty) = 0$$

and the performance index

$$I(u) = \frac{1}{2} \int_0^\infty (4x_1^2 + u^2) dt$$

Using the generic symbols of the example, in this numerical example

$$A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad \underline{b} = \begin{Bmatrix} 0 \\ 1 \end{Bmatrix}, \quad Q = \begin{bmatrix} 4 & 0 \\ 0 & 0 \end{bmatrix}, \quad \alpha = 1$$

Substituting these values in equation (A.2.32) results in

$$\begin{bmatrix} 2 - 2p_{12}^2 & p_{11} - 2p_{12}p_{22} \\ p_{11} - 2p_{12}p_{22} & 2p_{12} - 2p_{22}^2 \end{bmatrix} = 0 \qquad (A.2.34)$$

Here

$$P = \begin{bmatrix} p_{11} & p_{12} \\ p_{12} & p_{22} \end{bmatrix}$$

From (A.2.34)

$$p_{12} = 1 \quad , \quad p_{22} = 1 \text{ and } p_{11} = 2$$

Hence, from equation (A.2.30)

$$J = 2x_1^2 + 2x_1x_2 + x_2^2 \qquad (A.2.35)$$

and from equation (A.2.33), the control law is

$$u^* = -2x_1 - 2x_2 \qquad (A.2.36)$$

A.3 Pontryagin's Maximum Principle

From equations (A.2.15) and (A.2.16) it is evident that the optimal $\underline{u}$, i.e., $\underline{u}^*$, is such that it minimizes the Hamiltonian. This is the statement of the Pontryagin Maximum Principle. In other words

$$H(t, \underline{x}, \underline{u}^*, \underline{\lambda}) \leq H(t, \underline{x}, \underline{u}, \underline{\lambda}) \qquad (A.3.1)$$

for any $\underline{u} \neq \underline{u}^*$.

A.4 The Euler Equations - Unconstrained Control

A set of necessary conditions called the Euler Equations will be derived next for the optimization problem of Section A.1. In the following development it will be assumed that the control vector $\underline{u}(t)$ is unconstrained. The deviation here is different from the usual methods of the classical calculus of variations. It leans heavily on the functional equation of dynamic programming, viz., equation (A.2.14).

Equation (A.2.14) is equivalent to the two equations

216

$$\nabla_{\underline{u}} g + \underline{f}_{\underline{u}} \nabla_{\underline{x}} J = 0 \qquad\qquad (A.4.1)$$

which is the condition for the term within the square brackets
to be a minimum with respect to the vector $\underline{u}$ and

$$\frac{\partial J}{\partial t} + g(t, \underline{x}, \underline{u}) + < \underline{f}(t, \underline{x}, \underline{u}), \nabla_{\underline{x}} J > = 0 \qquad (A.4.2)$$

valid for t, $\underline{x}$ and $\underline{u}$ related by (A.4.1). Note that (A.4.1)
is a vector equation. In equation (A.4.1) $\nabla_{\underline{u}} g$ is the m-dimen-.
sional vector with components

$$\left(\frac{\partial g}{\partial u_1}, \frac{\partial g}{\partial u_2}, \cdots, \frac{\partial g}{\partial u_m} \right)'$$

and $\underline{f}_{\underline{u}}$ is the m x n matrix defined by

$$\underline{f}_{\underline{u}} = \begin{bmatrix} \dfrac{\partial f_1}{\partial u_1} & \dfrac{\partial f_2}{\partial u_1} & \cdots & \dfrac{\partial f_n}{\partial u_1} \\ \vdots & & & \\ \dfrac{\partial f_1}{\partial u_m} & \dfrac{\partial f_2}{\partial u_m} & \cdots & \dfrac{\partial f_n}{\partial u_m} \end{bmatrix} \qquad\qquad (A.4.3)$$

Now consider

$$\frac{d}{dt}\left(\nabla_{\underline{x}}J\right) = \left(\nabla_{\underline{xx}}J\right)\dot{\underline{x}} + \nabla_{\underline{x}}\frac{\partial J}{\partial t} \qquad (A.4.4)$$

where

$$\nabla_{\underline{xx}} \triangleq \begin{bmatrix} \dfrac{\partial^2}{\partial x_1^2} & \dfrac{\partial^2}{\partial x_1 \partial x_2} & \cdots & \dfrac{\partial^2}{\partial x_1 \partial x_n} \\[2ex] \vdots & & & \\[1ex] \dfrac{\partial^2}{\partial x_n \partial x_1} & \cdots\cdots\cdots & & \dfrac{\partial^2}{\partial x_n^2} \end{bmatrix} \qquad (A.4.5)$$

Taking the expression for the gradient on both sides of equation (A.4.2) w. r. t. $\underline{x}$ yields

$$\nabla_{\underline{x}}\frac{\partial J}{\partial t} + \nabla_{\underline{x}}g + \left(\nabla_{\underline{xx}}J\right)\underline{f} + \underline{f}_{\underline{x}}\left(\nabla_{\underline{x}}J\right) = 0 \qquad (A.4.6)$$

i.e.,

$$\left(\nabla_{\underline{xx}}J\right)\underline{f} + \nabla_{\underline{x}}\frac{\partial J}{\partial t} = -\nabla_{\underline{x}}g - \underline{f}_{\underline{x}}\left(\nabla_{\underline{x}}J\right) \qquad (A.4.7)$$

Replacing $\dot{\underline{x}}$ by $\underline{f}$ in equation (A.4.4) and then substituting in (A.4.7) yields

$$\frac{d}{dt} (\nabla_{\underline{x}} J) = - \nabla_{\underline{x}} g - \underline{f}_{\underline{x}} (\nabla_{\underline{x}} J) \qquad (A.4.8)$$

Define

$$\nabla_{\underline{x}} J = \underline{\lambda} \qquad (A.4.9)$$

From (A.4.8) and (A.4.9)

$$\dot{\underline{\lambda}} = - \nabla_{\underline{x}} g - \underline{f}_{\underline{x}} \underline{\lambda} \qquad (A.4.10)$$

and from (A.4.1) and (A.4.9)

$$\nabla_{\underline{u}} g + \underline{f}_{\underline{u}} \underline{\lambda} = 0 \qquad (A.4.11)$$

Equations (A.1.1), (A.4.10) and (A.4.11) are the Euler equations for the optimization problem. They represent a set of 2n first order differential equations and m finite equations involving 2n + m variables and hence can be solved when 2n boundary conditions are specified.

The 2n + m Euler equations can be combined and equivalently expressed as 2n first order differential equations in the so-called <u>Hamilton's canonic form.</u> This form makes use of the Hamiltonian defined in equation (A.2.16).

From equation (A.2.16), it is seen that

$$\nabla_{\underline{u}} H = \nabla_{\underline{u}} g + \underline{f}_{\underline{u}} \underline{\lambda} \qquad (A.4.12)$$

$$\nabla_{\underline{\lambda}} H = \underline{f}(t, \underline{x}, \underline{u}) \qquad (A.4.13)$$

$$\nabla_{\underline{x}} H = \nabla_{\underline{x}} g + \underline{f}_{\underline{x}} \underline{\lambda} \qquad (A.4.14)$$

Comparing equations (A.1.1), (A.4.10), and (A.4.11) with equations (A.4.12) to (A.4.14), it is seen that the Euler equations can be written in terms of the Hamiltonian in the following form

$$\dot{\underline{x}} = \nabla_{\underline{\lambda}} H \qquad (A.4.15)$$

$$-\dot{\underline{\lambda}} = \nabla_{\underline{x}} H \qquad (A.4.16)$$

$$0 = \nabla_{\underline{u}} H \qquad (A.4.17)$$

The solution of equation (A.4.17) is by definition equation (A.2.18). Hence if $\underline{u}^*$ is used instead of $\underline{u}$ in equations (A.4.15) and (A.4.16), equation (A.4.17) will automatically be satisfied. However, using $\underline{u}^*$ instead of $\underline{u}$ in the

right hand sides of equations (A.4.15) and (A.4.16) is equivalent to using H* of equation (A.2.19) instead of H. This leads to the <u>Hamilton</u> <u>canonic</u> <u>equations</u>

$$\dot{\underline{x}} = \nabla_{\underline{\lambda}} H^* \qquad \text{(A.4.18)}$$

$$- \dot{\underline{\lambda}} = \nabla_{\underline{x}} H^* \qquad \text{(A.4.19)}$$

Equations (A.4.18) and (A.4.19) are <u>necessary</u> conditions which have to be satisfied on an optimal trajectory. They represent a set of 2n first order differential equations. To obtain a solution to this set of equations, 2n boundary conditions are necessary. These conditions may be determined either from the transversality conditions to be discussed later or they may be specified beforehand by requiring that the trajectories should originate and terminate at certain points in the state space.

Finally

$$\frac{dH^*(t, \underline{x}, \underline{\lambda})}{dt} = \frac{\partial H^*}{\partial t} + < \dot{\underline{x}}, \nabla_{\underline{x}} H^* > + < \dot{\underline{\lambda}}, \nabla_{\underline{\lambda}} H > \qquad \text{(A.4.20)}$$

Substituting from equations (A.4.18) and (A.4.19) in

(A.4.20) yields

$$\frac{dH^*(t, \underline{x}, \underline{\lambda})}{dt} = \frac{\partial H^*(t, \underline{x}, \underline{\lambda})}{dt} \qquad (A.4.21)$$

Equation (A.4.21) shows that if H* does not depend explicitly on t (i.e., H* = H*$(\underline{x}, \underline{\lambda})$), then on an optimal trajectory, the "Hamiltonian" function is a constant, i.e.,

$$H^*(\underline{x}, \underline{\lambda}) = \text{constant} \qquad (A.4.22)$$

Example

Consider the example of the linear, time-invariant plant with quadratic integrand in the performance index and infinite process duration described by equations (A.2.22) and (A.2.23). For this system, the minimum value of the Hamiltonian is given by equation (A.2.27). From equation (A.2.27) and equations (A.4.18) and (A.4.19), the Hamilton's canonic equations are

$$\underline{\dot{x}} = A\,\underline{x} - \frac{1}{\alpha}\,\underline{b}\,\underline{b}^T\,\underline{\lambda} \qquad (A.4.23)$$

$$-\underline{\dot{\lambda}} = Q\,\underline{x} + A^T\,\underline{\lambda} \qquad (A.4.24)$$

Equations (A.4.23) and (A.4.24) represent a set of 2n

simultaneous linear time-invariant first order differential equations. These are to be solved with the given boundary conditions:

$$x_1(0) = C_1 \quad x_1(\infty) = 0$$

$$x_2(0) = C_2 \quad x_2(\infty) = 0$$

(A.4.25)

The solution yields the optimal trajectory $\underline{x}^*(t)$. The optimal control $u^*(t)$ is then determined using equation (A.2.26).

Now using the numerical values associated with the double integration plant of the previous example, equations (A.4.23) and (A.4.24) reduce to

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = -\lambda_2$$

(A.4.26)

$$\dot{\lambda}_1 = -4x_1$$

$$\dot{\lambda}_2 = -\lambda_1$$

Rewrite equations (A.4.26) compactly as

$$\dot{\underline{z}} = B\,\underline{z} \tag{A.4.27}$$

where $\underline{z}$ = column $(x_1,\ x_2,\ \lambda_1,\ \lambda_2)$ and the matrix B is

$$B = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0' & -1 \\ -4 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \end{bmatrix} \tag{A.4.28}$$

The characteristic equation of the canonical system

$$\det\,|\,B - \mu\,I\,| = 0$$

yields

$$\mu^4 + 4 = 0 \tag{A.4.29}$$

From equation (A.4.29), the eigenvalues are

$$\mu_1 = -1 + j\ ;\ \mu_2 = -1 - j\ ;\ \mu_3 = 1 + j\ ;\ \mu_4 = 1 - j$$

224

Hence the solution of the canonic equations is of the form

$$x_1(t) = k_1 \exp[(-1 + j)t] + k_2 \exp[(-1 - j)t] +$$

$$k_3 \exp[(1 + j)t] + k_4 \exp[(1 - j)t] \quad (A.4.30)$$

etc.

Note that the canonic system is unstable (half of its eigenvalues are in the right half plane). Hence the only way to satisfy the terminal boundary conditions in (A.4.25) is to make the constants associated with the response due to the right half plane eigenvalues zero.

This implies in equation (A.4.30)

$$k_3 = k_4 = 0 \quad (A.4.31)$$

Then

$$x_1 = k_1 \exp(\mu_1 t) + k_2 \exp(\mu_2 t)$$

$$(A.4.32)$$

$$x_2 = k_1 \mu_1 \exp(\mu_1 t) + k_2 \mu_2 \exp(\mu_2 t)$$

From the initial conditions in equation (A.4.25)

$$k_1 + k_2 = c_1$$

<div align="right">(A.4.33)</div>

$$\mu_1 k_1 + \mu_2 k_2 = c_2$$

k_1 and k_2 can now be explicitly determined.

From equation (A.2.26)

$$u^*(t) = - \lambda_2(t)$$

and hence from equation (A.4.26)

$$u^*(t) = \dot{x}_2$$
$$= k_1 \mu_1^2 \exp(\mu_1 t) + k_2 \mu_2^2 \exp(\mu_2 t) \qquad \text{(A.4.34)}$$

Equation (A.4.34) yields explicitly the optimal open loop control function. Thus, the optimum open loop solution of this particular optimization problem has been determined.

Remark 1

In general the canonic equations represent only nec-essary conditions for optimality. Hence the solution of the canonic equations yield, in general, only a "candidate" for the optimum control function. In this particular example it can be shown that the solution is actually optimum. In general

however, the solution has to satisfy a few other necessary
conditions.

Remark 2

In general, the solution of the canonic equations
yields the optimum open loop solution. The Hamilton-Jacobi
or the dynamic programming formulation of the optimum control
problem will yield the closed loop or "feedback law" solutions.

Remark 3

In the case of linear plants and quadratic performance
criteria, one half of the eigenvalues of the canonic equations
for the regulator problem will have negative real parts. Hence,
in the case of free terminal time, fixed terminal point prob-
lems, the so-called "transversality conditions" are satisfied
by making the optimum system asymptotically stable. In gen-
eral, however, the terminal conditions will represent a curve
or a surface in the solution space* - this surface is some-
times called the terminal manifold. (Such, for example, is
the case in missile interception problems, the rendezvous of
two space vehicles, etc). Hence one must show that the solu-
tion of the optimization problem reaches the terminal

---

*The solution space is the (n + 1) dimensional space
whose co-ordinates are the n state coordinates and time.

manifold, i.e., the optimal trajectory $\underline{x}^*(t)$ is non-tangential to the terminal manifold in the solution space. This condition of non-tangency is called the transversality condition. Satisfaction of the transversality condition guarantees that the optimal system will reach the target. This transversality condition is discussed in greater detail in the next section.

Remark 4

The canonic equations have been derived in this section using the assumption that the control function has no constraints. However, it can be shown that the canonic equations are necessary conditions even when there are constraints on the control function. Of course, when there are constraints, the minimization of the Hamiltonian with respect to the control vector would involve more than examining the set of equations obtained by setting the suitable partial derivatives of the Hamiltonian equal to zero. The minimum value may occur on the boundary of the allowable region in which the control vector is constrained to lie.

A.5 The Transversality Condition

Suppose now that the trajectory $\underline{x}(t)$ must terminate on a given manifold $\underline{x} = \underline{h}(t)$. In this case, for the optimal curve, the change in $J(\underline{x}, t)$, the return function, as the final point

228

moves along the specified curve must be zero. This is equiv-
alent to saying that at the final point

$$\frac{\partial J}{\partial t} + < \nabla_{\underline{x}} J, \, \underline{\dot{h}} > = 0 \qquad (A.5.1)$$

Combining this with equation (A.4.2) yields the condition
at the final point

$$g(t, \underline{x}, \underline{u}) + < \underline{f}(t, \underline{x}, \underline{u}), \nabla_{\underline{x}} J > - < \nabla_{\underline{x}} J, \, \underline{\dot{h}} > = 0 \qquad (A.5.2)$$

which combined with equation (A.4.9) yields

$$g(t, \underline{x}, \underline{u}) - < \underline{\dot{h}} - \underline{f}(t, \underline{x}, \underline{u}), \, \underline{\lambda} > = 0 \qquad (A.5.3)$$

at the final point.

Equation (A.5.3) is usually written in the form

$$[g(t, \underline{x}, \underline{u}) + < \underline{\lambda}, \, \underline{f}(t, \underline{x}, \underline{u}) >] \Big|_{t=T} dt - < \underline{\lambda}(T), \, d\underline{x} > = 0$$

$$(A.5.4)$$

where $\underline{\dot{h}}$ is replaced by $d\underline{x}/dt$. In equation (A.5.4) $d\underline{x}$ and dt
are differentials on the terminal manifold at the point of its
intersection with the optimal trajectory. Note that the other
terms in equation (A.5.4) are evaluated on the optimal trajec-
tory.

Equation (A.5.4) is the transversality condition. Note
that in equation (A.5.4) the $\underline{x}$, $\underline{u}$ and $\underline{\lambda}$ refer to their values

corresponding to the optimal trajectory.

In terms of the Hamiltonian, equation (A.2.19), the transversality condition equation (A.5.4) is equivalent to

$$H^*(t, \underline{x}, \underline{\lambda}) \Big|_{t=T} dt - <\underline{\lambda}(T), d\underline{x}> = 0 \qquad (A.5.5)$$

It is again emphasized that in equation (A.5.5), dt and d$\underline{x}$ are differentials on the terminal manifold.

In the case of a fixed time optimization problem with terminal state free, equation (A.5.5) requires that

$$\underline{\lambda}(T) = \underline{0} \qquad (A.5.6)$$

In the case of a fixed time optimization problem with the final state specified, equation (A.5.5) is automatically satisfied since d$\underline{x}$ = $\underline{0}$ and dt = 0.

A.6 A Minimum Time Problem - Use of Maximum Principle

Consider a two-integration second order plant which is to be brought to the equilibrium state (= the origin of the state space) in minimum time. The plant equations are

$$\dot{x}_1 = x_2 \; ; \; \dot{x}_2 = u \qquad |u| \leq 1$$

The performance index is

$$I = \int_0^T dt \quad (T \text{ free}), \quad x_1(T) = x_2(T) = 0 .$$

The 'Hamiltonian' for this problem is

$$H(\underline{x}, \underline{\lambda}, u) = \lambda_1 x_2 + \lambda_2 u + 1$$

Note that $\partial H/\partial u = \lambda_2$ for any control u. Hence the 'Hamiltonian' attains its minimum value on the boundary of the admissible controls, i.e.,

$$u^* = - \operatorname{sgn} \lambda_2 = \begin{cases} -1 & \lambda_2 > 0 \\ 0 & \lambda_2 = 0 \\ +1 & \lambda_2 < 0 \end{cases}$$

and

$$H^*(\underline{x}, \underline{\lambda}) = \lambda_1 x_2 - \lambda_2 \operatorname{sgn} \lambda_2 + 1$$

The canonical equations are

$$\dot{x}_1 = x_2 \qquad \dot{\lambda}_1 = 0$$

$$\dot{x}_2 = - \operatorname{sgn} \lambda_2 \qquad \dot{\lambda}_2 = - \lambda_1$$

Since the terminal conditions are $x_1(T) - x_2(T) = 0$ and

the transversality condition (A.5.5) yields $\lambda_2(T) = \pm 1$,

one can now solve the canonical equations for the optimal

control function $u^*(t) = -\,\text{sgn}\,\lambda_2(t)$ for any initial state

$[x_1(0),\ x_2(0)]$. The result is of the form shown in Fig. A.2.

Note that the control here is of the relay ("bang-bang",

or "on-off") type. This result has been first obtained by
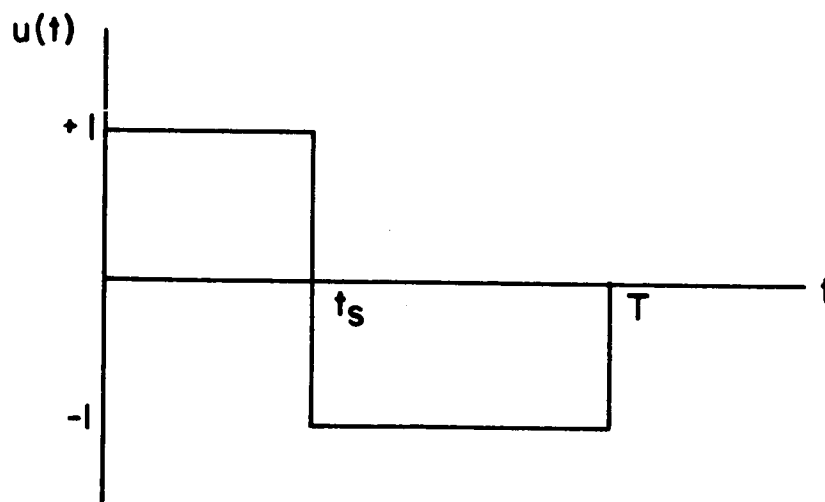
Bushaw [19].

FIGURE A.2

It is interesting to note the simplicity with which this

result is obtained from the maximum principle.

So far nothing has been said about the solutions of the

canonical equations or the validity of the Maximum Principle

(A.3.1), at corners. A continuity argument can be used to

show the validity of (A.3.1) even at corners of an optimal

trajectory. It can also be shown that the functions $\underline{x}^*(t)$,

$\underline{\lambda}^*(t)$ and $H^*(t, \underline{x}^*(t)), \underline{\lambda}^*(t))$ are continuous functions of t.

The latter statement is equivalent to the <u>Weierstrass-Erdmann</u>

<u>corner</u> <u>condition</u> in the calculus of variations.

A.7 The Solution of a Discrete-Time Optimization Problem

In this section, the application of Bellman's functional

equation to discrete-time (or sampled-data) systems will be

pointed out. This approach was first used by Kalman [20]

for solving optimization problems in linear sampled-data

systems, and Kalman's results are reviewed here.

Consider the optimal control problem in discrete time.

Differential equations are replaced by difference equations

and integrals by sums. It should be noted that any optimiza-

tion problem that is to be solved on a digital computer must

be discretized initially. Also, when the limit is taken, as

the sampling period goes to zero, the continuous optimization

problem should result. On this basis, the functional equa-

tion of Bellman represents a general approach to the actual

numerical solution of optimization problems. Starting with

Bellman's equation both discrete and as already discussed,

continuous time optimization problems may be considered in a

systematic manner.

For the discrete-time case, consider the performance index to be

$$I_N = \sum_{i=0}^{N-1} g(\underline{x}(i), \underline{u}(i), i) \qquad (A.7.1)$$

Since in this case, choice of the initial and final control signals $\underline{u}(0)$ and $\underline{u}(N-1)$ doesn't affect $\underline{x}(0)$ and $\underline{x}(N-1)$, it is necessary to write the performance index for the N-stage control process as

$$I_N = \sum_{i=0}^{N-1} g(\underline{x}(i+1), \underline{u}(i), i+1) \qquad (A.7.2)$$

or

$$I_N = \sum_{i=1}^{N} g\left[\underline{x}(i), \underline{u}(i-1), i\right]$$

one is required to find the control sequence $\underline{u}(0), \underline{u}(1), \ldots, \underline{u}(N-1)$ that minimizes (A.7.2) under the constraints

$$\underline{x}(k+1) = \Phi(k) \underline{x}(k) + H(k) \underline{u}(k)$$

$$\underline{x}(0) = \underline{c} \qquad (A.7.3)$$

234

$$\underline{x}(N) = \text{free}$$

Problems similar to this were discussed in the previous sections. The problem with $\underline{x}(N)$ free and N fixed will be considered as an example.

With $\underline{x}(0)$ fixed, the minimum value of (A.7.2) is a function only of the initial state and the length of the process. Therefore let

$$J_N[\underline{x}(0)] = \min_{\underline{u}(0),\ \underline{u}(1),\ \ldots,\ \underline{u}(N-1)} I_N[\underline{x},\ \underline{u},\ i] \qquad (A.7.4)$$

The basic functional equation is obtained by the following reasoning. Since each stage of the N stage process must be optimum, assume one is faced with the selection of the first control signal $\underline{u}(0)$. Any choice of $\underline{u}(0)$ say $\underline{u}'(0)$, will result in

$$I_N = g(\underline{x}(1),\ \underline{u}'(0),\ 1) + J_{N-1}[\underline{x}(1)] \qquad (A.7.5)$$

The second term on the right is present since the process must be optimum for the remaining N - 1 stages. Since $\underline{x}(1)$ is a function of $\underline{u}(0)$ (by (A.7.3)), the minimum value of the performance index is obtained by minimizing (A.7.5) with respect

to $\underline{u}'(0)$. Thus

$$J_N[\underline{x}(0)] = \min_{\underline{u}(0)} \left[g(\underline{x}(1), \underline{u}(0), 1) + J_{N-1}[\underline{x}(1)] \right] \quad \text{(A.7.6)}$$

Equation (A.7.6) is the basic functional equation for the problem under consideration. Iterative solution of this equation yields the required sequence $\underline{u}(0)$, $\underline{u}(1)$, ..., $\underline{u}(N-1)$.

An example of this procedure is presented at this time.

Problem Statement

Given a linear time-invariant plant subject to a piecewise constant input signal

$$\underline{x}(k+1) = \Phi \underline{x}(k) + \underline{h}\, u(k) \quad \text{(A.7.7)}$$

$$u(k) = \text{constant} \quad t_k \le t < t_{k+1} \quad \text{(A.7.8)}$$

Determine the control law

$$u = u(\underline{x})$$

that minimizes the performance index

$$I_N(\underline{x}(0), u) = \sum_{i=1}^{N} \underline{x}(i)'\, Q\, \underline{x}(i) + \alpha\, u(i-1)^2 \quad \text{(A.7.9)}$$

236

Problem Solution

Define

$$J_N[x(0)] = \min_u I_N[\underline{x}(0), u] \qquad (A.7.10)$$

By using the principle of optimality, one knows that at the beginning of the N-stage process one must make an optimal decision. Choosing any value of u(0) will result in

$$I_1[x(0), u(0)] \qquad (A.7.11)$$

Since the remaining N - 1 stages must constitute an optimal policy, the performance index is

$$I_N(\underline{x}(0), u) = I_1[\underline{x}(0), u(0)] + J_{N-1}[\underline{x}(1)] \qquad (A.7.12)$$

In order to minimize this function over the total N stages, write

$$J_N[\underline{x}(0)] = \min_{u(0)} [I_1(\underline{x}(0), u(0)) + J_{N-1}(\underline{x}(1))] \qquad (A.7.13)$$

which is the same as (A.7.6). To start the solution, one requires that when N = 1

$$J_1[\underline{x}(0)] = \min_{u(0)} I_1(\underline{x}(0), u(0)) \qquad \text{(A.7.14)}$$

The following procedure has been rigorously justified [20]. Consider the matrix Q to be symmetric and $\alpha \geq 0$. The optimal return from an N stage process starting at $\underline{x}(0)$ may be written as

$$J_N[\underline{x}(0)] = \underline{x}'(0) \ P(N) \ \underline{x}(0) \qquad \text{(A.7.15)}$$

where $P(N)$ is symmetric. Using this in (A.7.13), one has

$$J_N[\underline{x}(0)] = \min_{u(0)} [\underline{x}'(1) \ Q \ \underline{x}(1) + \alpha \ \jmath(0)^2 + \underline{x}'(1) \ P(N-1) \ \underline{x}(1)]$$
$$\text{(A.7.16)}$$

and from (A.7.7)

$$J_N[\underline{x}(0)] = \min_{u(0)} [\underline{x}'(0) \ \Phi' \ [Q + P(n - 1)] \ \Phi \ \underline{x}(0)$$

$$+ 2\underline{h}' \ [Q + P(N - 1)] \ \Phi \ \underline{x}(0) \ u(0)$$

$$+ [\underline{h}' \ [Q + P(N - 1)] \ \underline{h} + \alpha] \ u^2(0)] \qquad \text{(A.7.17)}$$

From (A.7.17), define

238

$$S(N - 1) \overset{\Delta}{=} Q + P(N - 1) \qquad \text{(A.7.18)}$$

The value of u(0) that minimizes (A.7.17) is easily found to be

$$u(0) = - \frac{\underline{h}' \ S(N - 1) \ \Phi \ \underline{x}(0)}{\underline{h}' \ S(N - 1) \ \underline{h} + \alpha} = \underline{f}'(N - 1) \ \underline{x}(0) \qquad \text{(A.7.19)}$$

From (A.7.19), u(0) is then that value of control signal to be used at the start of an N-stage process. For this reason a subscript N will be included and (A.7.19) will be written as

$$u_N(0) = \underline{f}'(N - 1) \ \underline{x}(0) \qquad \text{(A.7.20)}$$

Notice that (A.7.20) represents a linear combination of the state variables.

The minimum value of (A.7.17) may be written as

$$J_N[\underline{x}(0)] = \underline{x}'(0) \ P(N) \ \underline{x}(0)$$

$$= \underline{x}'(0) \ [\Phi + \underline{h} \ \underline{f}'(N - 1)]' \ S(N - 1) \ [\Phi + \underline{h} \ \underline{f}'(N - 1)] \ \underline{x}(0)$$

$$+ \alpha \ \underline{x}'(0) \ \underline{f}(N - 1) \ \underline{f}'(N - 1) \ \underline{x}(0) \qquad \text{(A.7.21)}$$

Thus from (A.7.21) one has a recurrence equation for $P(N)$,

$$P(N) = [\Psi + \underline{h}\ \underline{f}'(N - 1)]'\ S(N - 1)[\Psi + \underline{h}\ \underline{f}'(N - 1)]$$

$$+ \alpha\ \underline{f}(N - 1)\ \underline{f}'(N - 1)$$

$$(A.7.22)$$

It is now possible to solve for the optimum control law in an iterative manner as follows: Start with a one stage process (i.e., $N = 1$). From (A.7.16) and (A.7.22)

$$J_1[\underline{x}(0)] = \min_{u(0)}\ [\underline{x}'(1)\ Q\ \underline{x}(1) + \alpha\ u^2(0)]$$

$$= \min_{u(0)}\ [\underline{x}'(1)[Q + P(0)]\ \underline{x}(1) + \alpha\ u^2(0)] \qquad (A.7.23)$$

Therefore let $P(0) = 0$ and from (A.7.18) $S(0) = Q$. Using (A.7.19),

$$u_1(0) = \underline{f}'(0)\ \underline{x}(0) \qquad (A.7.24)$$

In order to obtain $u_2(0)$ (i.e., the first signal for a two stage process), use $\underline{f}'(0)$ obtained in (A.7.24) with (A.7.21) in order to calculate $P(1)$. When $P(1)$ is calculated, use (A.6.18) to determine $S(1)$. Equation (A.7.19) is then used to calculate $u_2(0)$ and the calculations unfold in this manner. The optimum

feedback coefficients are obtained as

$$\underline{f}'(0), \underline{f}'(1), \underline{f}'(2), \ldots, \underline{f}'(N-1) \qquad \text{(A.7.25)}$$

These coefficients are then used in reverse order. For example, for a three stage process,

$$u_1(0) = \underline{f}'(0) \underline{x}(2)$$

$$u_2(0) = \underline{f}'(1) \underline{x}(1) \qquad \text{(A.7.26)}$$

$$u_3(0) = \underline{f}'(2) \underline{x}(0)$$

Equation (A.7.26) has the following meaning. $u_1(0)$ is the optimal _first_ signal for a 1 stage process; $u_2(0)$ is the optimal _first_ signal for a 2 stage process and $u_3(0)$ is the optimal _first_ signal for a three stage process. Therefore the feedback coefficients for the three stage process are: at $t = 0$, multiply the state of the plant by $\underline{f}'(2)$, at $t = T$, by $\underline{f}'(1)$ and at $t = 2T$ by $\underline{f}'(0)$. Notice that in general, the control law is non-stationary for finite length control processes. It has been shown in general that as the number of stages in the process approaches infinity, the value of $\underline{f}'(\infty)$ approaches a constant and a linear, time-invariant control law results. In practice, the feedback coefficients generally converge rather

rapidly and approximately optimal control of finite length processes may be obtained by using constant feedback from the state variables.

A.8. Conclusions

Modern optimal control theory marks a distinct departure in philosophy as well as in method from classical control techniques. First, the object is no longer merely to come within a given set of specifications but rather to go further, in fact to go all the way to an optimum solution. In order to accomplish this feat a great deal of information must be given about the plant and its desired performance. A second difference is the need for an index of performance in the modern theory, which requires the designer to completely specify desirable performance as a function. In truth it must be admitted that this is difficult or impossible to do with the present state of knowledge. Further study in this area is surely needed. Moreover this is not the type of study which can be made by researchers unfamiliar with applications. A vast backlog of engineering experience with a variety of performance indices appears to be the only way out of this dilemma.

It should be noted that no attempt has been made in this appendix to discuss the problems associated with hard constraints. A detailed consideration of the problem of constrained control is given in chapter 4 of this report.

Appendix B

Method of Quasi-Linearization

Let a vector differential equation

$$\dot{\underline{x}} = \underline{f}(\underline{x}, t) \qquad t_0 \geq t \geq t_T \qquad (B.1)$$

be given with the boundary conditions

$$< \underline{c}(t_i), \underline{x}(t_i) > = b_i \qquad i = 1, 2, \ldots, n \qquad (B.2)$$

$$t_0 \leq t_1 \leq \ldots \leq t_n \leq t_T$$

where $\underline{c}$ and $\underline{x}$ are n-dimensional vectors. It is assumed that equations (B.1) and (B.2) have a unique solution on $[t_0, t_T]$.

Let $\underline{x}_0(t)$ be an initial guess to the solution of equation (B.1) on $[t_0, t_T]$. The $(k + 1)$-st approximation is then obtained from the k-th via

$$\dot{\underline{x}}_{k+1} = \underline{f}(\underline{x}_k, t) + J(\underline{f}(\underline{x}_k, t))(\underline{x}_{k+1} - \underline{x}_k) \qquad (B.3)$$

and $\underline{x}_{k+1}$ satisfies equation (B.2), where J is the Jacobian matrix whose ij-th element, $\partial f_i / \partial x_j$, is the partial derivative of the ith component of $\underline{f}$ with respect to the jth component

243

of $\underline{x}$.

The components of the initial approximation vector $\underline{x}_0(t)$ may be constants, suitably chosen functions of time, polynomials in t, etc. The first approximation $\underline{x}_1(t)$ is obtained as the solution of

$$\underline{\dot{x}}_1 = \underline{f}(\underline{x}_0, \ t) + J(\underline{f}(\underline{x}_0, \ t))(\underline{x}_1 - \underline{x}_0) \tag{B.4}$$

$$= J(\underline{f}(\underline{x}_0, \ t)) \ \underline{x}_1 + f(\underline{x}_0, \ t) - J(\underline{f}(\underline{x}_0, \ t))\underline{x}_0 \tag{B.5}$$

satisfying equation (B.2).

Let $\Phi_1(t)$ be the fundamental solution matrix of

$$\dot{\Phi}_1 = J(\underline{f}(\underline{x}_0, \ t)) \ \Phi_1, \qquad \Phi_1(0) = \text{identity matrix} \tag{B.6}$$

Let $\underline{p}_1(t)$ be the particular solution vector of

$$\underline{\dot{p}}_1 = J(\underline{f}(\underline{x}_0, \ t)) \ \underline{p}_1 + f(\underline{x}_0, \ t) - J(\underline{f}(\underline{x}_0, \ t))\underline{x}_0,$$

$$\underline{p}_1(0) = \underline{0} \tag{B.7}$$

Then the solution of equation (B.5) is written as

$$\underline{x}_1(t) = \Phi_1(t) \ \underline{k}_1 + \underline{p}_1(t) \tag{B.8}$$

where $\underline{k}_1$ is a constant vector determined by solving

$$< \underline{c}(t_i), \ (\Phi_1(t_i)\underline{k}_1 + \underline{p}_1(t_i)) \ > = b_i \quad i = 1, \ 2, \ \ldots, \ n$$

$$(B.9)$$

The entire calculations are easily carried out on a digital computer. The convergence of this scheme, which is quadratic in nature, and many other problems are discussed in reference [10], [22].

Method of Differential Approximation

An interesting problem which has many practical applications is the following: Given a vector valued function $\underline{\varphi}(t)$ of dimension n defined in the interval $0 \leq t \leq T$, is it possible to find an n-dimensional vector differential equation of the form

$$\underline{\dot{x}} = \underline{f}(t, \underline{x}) \qquad (C.1)$$

such that the solution of this differential equation with initial conditions

$$\underline{x}(0) = \underline{\varphi}(0) \qquad (C.2)$$

is identical with $\underline{\varphi}(t)$ over the interval $0 \leq t \leq T$?

The solution to this problem is rather difficult to find in general. However, a slightly reformulated version of this problem is rather easy to solve and quite adequate in practice. The reformulated problem is posed as follows: Again, given the function $\underline{\varphi}(t)$ defined above and the differential equation.

$$\underline{\dot{x}} = \underline{g}(t, \underline{x}, \underline{b}) \qquad (C.3)$$

where the form of the function $\underline{g}$ is known except for a finite set of parameters $\underline{b}$, determine $\underline{b}$ such that the solution of (C.3) with initial conditions (C.2) is "closest" to $\underline{\varphi}(t)$, over the interval $0 \leq t \leq T$ the term closest being suitably defined. Note that if a set $\underline{b} = \underline{b}_1$ existed such that

$$\dot{\underline{\varphi}}(t) = \underline{g}(t, \underline{\varphi}, \underline{b}_1) \quad 0 \leq t \leq T \tag{C.4}$$

then, this is the set which will make the solution of the diferential equation (C.3) with initial condition (C.2) identical with $\underline{\varphi}(t)$. However, in general such a set of parameters will not exist.

A reasonable compromise is to seek for a set $\underline{b} = \underline{b}_2$ such that a suitable function

$$\dot{\underline{\varphi}}(t) - \underline{g}(t, \underline{\varphi}, \underline{b}_2) \tag{C.5}$$

is close to zero in an acceptable sense. For example $\underline{b}_2$ may be obtained as the solution of

$$\underset{\underline{b}}{\text{Min}} \int_0^T || \dot{\underline{\varphi}}(t) - \underline{g}(t, \underline{\varphi}, \underline{b}) ||^2 \, dt \tag{C.6}$$

or

$$\underset{\underline{b}}{\text{Min}} \; \underset{0 \le t \le T}{\text{sup}} \quad || \underline{\dot{\varphi}}(t) - \underline{g}(t, \underline{\varphi}, \underline{b}_2) || \qquad (C.7)$$

In (C.6) and (C.7) $||\cdot||$ is the Euclidean norm.

The minimization problem implied by (C.6) is often easily solved by equating to zero the partial derivatives of the integral with respect to the components of $\underline{b}$, this yielding a sufficient set of simultaneous equations involving the components of $\underline{b}$, subsequently solving these simultaneous equations.

The minimization problem implied by (C.7) is much more difficult to solve.

The technique by which a set of parameters in a differential equation are selected so as to match its trajectory with a given function of time is called differential approximation [23].

APPENDIX D

The technique presented here of applying invariant imbedding to boundary value problems is essentially that of reference [29]. Consider the TPBVP described by the differential equations

$$\dot{x} = f(t, x, y)$$

$$\dot{y} = g(t, x, y) \tag{D.1}$$

with boundary conditions

$$y(0) = a \qquad y(T) = b \tag{D.2}$$

Let $r(C, T)$ denote the missing terminal condition on x for a process starting at time 0 and ending at time T and also satisfying $y(0) = a$, $y(T) = C$, i.e.,

$$x(T) = r(C, T) \tag{D.3}$$

In equation (D.3) C and T are regarded as independent variables. From equation (D.1) then

$$r(C + \Delta C, T + \Delta T) = r(C, T) + f(T, r, C)\Delta T + 0(\Delta^2) \tag{D.4}$$

where $\lim_{\Delta \to 0} \dfrac{0(\Delta^2)}{\Delta} = 0.$

Expanding the left hand side of equation (D.4) using Taylor's formula yields

$$r(C + \Delta C, T + \Delta T) = r(C, T) + \Delta C \frac{\partial r}{\partial C} + \Delta T \frac{\partial r}{\partial T} + 0(\Delta^2)$$

(D.5)

From equation (D.1)

$$\Delta C = g(T, r, C)\Delta T + 0(\Delta^2)$$

(D.6)

Equating the right hand sides of equations (D.4) and (D.5) and passing to the limit as $\Delta T \to 0$ yields

$$\frac{\partial r}{\partial T} + g(T, r, C) \frac{\partial r}{\partial C} = f(T, r, C)$$

(D.7)

Equation (D.7) is a partial differential equation which with the proper boundary conditions on r governs the dependence of the missing terminal conditions on x as a function of the duration of the process and the terminal conditions on y.

APPENDIX E

Rewriting equation (5.6.3)

$$\frac{\partial r}{\partial T} - \frac{\partial r}{\partial C} \frac{\partial H^*}{\partial r} (T, r, C) = \frac{\partial H^*}{\partial C} (T, r, C) \qquad (E.1)$$

substituting into equation (E.1) using equation (5.5.3) yields

$$\frac{\partial r}{\partial T} - \frac{\partial r}{\partial C} \left[Cg_r (T, r) - 2h_r (T, r) \{y - h(T, r)\} \right.$$

$$\left. + \frac{1}{2} \cdot \frac{C^2 \, w_r (T, r)}{w^2 (T, r)} \right] = g(T, r) - \frac{C}{2w(T, r)} \qquad (E.2)$$

where $g_r = \frac{\partial g}{\partial r}$.

Try an approximate solution for $r(C, T)$ of the form

$$r(C, T) = P(T)C + \hat{x}(T) \qquad (E.3)$$

Substituting equation (E.3) into equation (E.2) and expanding the result about $r(0, T)$ gives to first order

$$\frac{dP}{dT} C + \frac{d\hat{x}}{dT} + P(T) \left[-C \{g_{\hat{x}} (T, \hat{x}) + g_{\hat{x}\hat{x}} (T, \hat{x}) PC \} \right.$$

$$+ 2h_{\hat{x}} (T, \hat{x}) \{y - h(T, \hat{x})\} +$$

$$+ 2 \frac{\partial}{\partial \hat{x}} \{h_{\hat{x}}(T, \hat{x}) \ (y - h(T, \hat{x}))\} PC - \frac{1}{2} \frac{c^2 \ w_{\hat{x}}(T, \hat{x})}{w^2(T, \hat{x})}$$

$$- \frac{1}{2} c^2 \frac{\partial}{\partial \hat{x}} \{ \frac{w_{\hat{x}}(T, \hat{x})}{w^2(T, \hat{x})} \} PC] \qquad \text{(E.4)}$$

$$= g(T, \hat{x}) + g_{\hat{x}}(T, \hat{x}) PC - \frac{c}{2w(T, \hat{x})} - \frac{c}{2} \frac{\partial}{\partial \hat{x}} \{ \frac{1}{w(T, \hat{x})} \} PC$$

Collecting terms of order c^0, and those of order c^1, c^2, and c^3 yields

$$\frac{d\hat{x}}{dT} = g(T, \hat{x}) - 2P(T) h_{\hat{x}}(T, \hat{x}) \ (y - h(T, \hat{x}))$$

$$C \frac{dP}{dT} = 2CPg_{\hat{x}} - 2P \frac{\partial}{\partial \hat{x}} \{h_{\hat{x}}(y - h(T, \hat{x}))\} PC$$

$$- \frac{c}{2w(T, \hat{x})} + \text{(terms of order } c^2 \text{ and } c^3) \qquad \text{(E.5)}$$

If equation (E.5) is satisfied then so is equation (E.4). Dividing the P equation in (E.5) by C, substituting -P for P, and noting that only those solutions for which C = 0 are of interest, then the sequential estimator equations become

$$\frac{d\hat{x}}{dT} = g(T, \hat{x}) + 2P(T) h_{\hat{x}}(T, \hat{x}) \{y - h(T, \hat{x})\}$$

$$\frac{dP}{dT} = 2Pg_{\hat{x}}(T, \hat{x}) + 2P \frac{\partial}{\partial \hat{x}} \left[h_{\hat{x}}(T, \hat{x}) \{y - h(T, \hat{x})\} \right] P$$

$$+ \frac{1}{2w(T, \hat{x})} \qquad\qquad (E.6)$$

APPENDIX F

For the vector case consider the class of systems defined by

$$\dot{x} = g(t, x) + k(t, x) u \tag{F.1}$$

$$y(t) = h(t, x) + (\text{observation error})$$

where x is an n-vector

   $g(t, x)$ is an n-vector function

   $k(t, x)$ is an n x p vector function

   u is a p-vector unknown input

   $h(t, x)$ is an m-vector

   y is an m-vector output

Define the vector residual errors as

$$e_1(t) = y - h(t, \hat{x}) \tag{F.2}$$

$$e_2(t) = \dot{x} - g(t, \hat{x}) \tag{F.3}$$

The least estimate of $x(t)$ $0 \le t \le T$ is given by

$$\frac{\text{Min}}{x(t)} \; 0 \le t \le T \int_{0}^{T} \left[||e_1(t)||_Q^2 + ||e_2(t)||_W^2 \right] dt \qquad \text{(F.4)}$$

where $||\cdot||_Q^2$ and $||\cdot||_W^2$ are suitably defined quasi-norms. Denote by $\hat{x}(t)$ $0 \le t \le T$ the function which minimizes the expression (F.4). The least squares estimate of $x(T)$ is then $\hat{x}(T)$.

Using equation (F.1) for motivation and substituting from equations (F.2) and (F.3) into the expression (F.4), then minimizing the expression (F.4) with respect to $\bar{x}(t)$ $0 \le t \le T$ is equivalent to minimizing

$$\int_{0}^{T} \left[||y - h(t, \bar{x})||_Q^2 + ||\bar{u}||_{k'Wk}^2 \right] dt \qquad \text{(F.5)}$$

with respect to $\bar{x}(t)$ and $\bar{u}(t)$ $0 \le t \le T$ subject to the differential constraint

$$\dot{\bar{x}} = g(t, \bar{x}) + k(T, \bar{x})\bar{u} \qquad \text{(F.6)}$$

Let

$$V(t, \bar{x}) = k'(t, \bar{x}) W(t, \bar{x}) k(t, \bar{x}) \qquad \text{(F.7)}$$

and define the "pre-Hamiltonian" $H(t, \bar{x}, \lambda, \bar{u})$ by

$$H(t, \bar{x}, \lambda, \bar{u}) = ||y - h(t, \bar{x})||_Q^2 + ||\bar{u}||_V^2$$

$$+ \langle \lambda, g(t, \bar{x}) + k(t, \bar{x})\bar{u} \rangle \qquad \text{(F.8)}$$

where $\langle \cdot, \cdot \rangle$ denotes the Euclidian inner product. Setting

$$H_{\bar{u}} = \begin{bmatrix} H_{\bar{u}_1} \\ H_{\bar{u}_2} \\ \cdot \\ \cdot \\ \cdot \\ H_{\bar{u}_p} \end{bmatrix} = 0 \qquad \text{(F.9)}$$

solving for $\bar{u}(t, \bar{x}, \lambda)$ assuming V is not singular, and sub-stituting $\bar{u}$ back into H leads to the Hamiltonian $H^*(t, x^*, \lambda)$. The variable x^* replaces $\bar{x}$ to indicate that x^* is the tra-jectory along which the maximum principle is satisfied. The Hamiltonian is then

$$H^*(t, x^*, \lambda) = ||y - h(t, x^*)||_Q^2 + \langle \lambda, g(t, x^*) \rangle$$

$$- \frac{1}{4} \langle \lambda, k \, V^{-1} \, k' \, \lambda \rangle \qquad \text{(F.10)}$$

The Euler-Lagrange equations are then

256

$$\dot{x}^* = \frac{\partial H^*}{\partial \lambda} \ (t, \ x^*, \ \lambda)$$

(F.11)

$$\dot{\lambda} = -\frac{\partial H^*}{\partial x^*} \ (t, \ x^*, \ \lambda)$$

since T has been fixed, and $x^*(0)$ and $x^*(T)$ are free, the transversality conditions yield

$$\lambda(0) = 0 \qquad\qquad \lambda(T) = 0 \qquad\qquad (F.12)$$

In order to solve the sequential estimation problem it is necessary to solve the TPBVP given by equations (F.11) with boundary conditions (F.12) for all T, where now the variable T is regarded as an independent variable.

Imbedding these TPBVP's in a larger class of TPBVP's with boundary conditions

$$\lambda(0) = 0 \qquad\qquad \lambda(T) = C \qquad\qquad (F.13)$$

and letting the missing terminal condition on x be $r(C, T)$, then $r(C, T)$ satisfies the invariant imbedding equation

$$\frac{\partial r}{\partial T} - \frac{\partial r}{\partial C} \frac{\partial H^*}{\partial r} \ (T, \ r, \ C) = \frac{\partial H^*}{\partial C} \ (T, \ r, \ C) \qquad (F.14)$$

where $\quad \dfrac{\partial r}{\partial C} = \left[\ \dfrac{\partial r_i}{\partial C_j} \ \right]$

Consider an approximate solution of the non-linear partial differential equation (F.14) of the form

$$r(C, T) = P(T)C + \hat{x}(T) \qquad (F.15)$$

where $P(T)$ is an $n \times n$ matrix

C and $\hat{x}$ are n-vectors

Substituting equation (F.15) into equation (F.14) gives

$$\frac{dP}{dt} C + \frac{d\hat{x}}{dT} - P(T) \frac{\partial H^*}{\partial r} (T, PC + \hat{x}, C) = \frac{\partial H^*}{\partial C}(T, PC + \hat{x}, C)$$

$$(F.16)$$

Now expand equation (F.16) about $r(0, T)$ retaining terms to first order. The motivation for this approach is that only those solutions of equation (F.14) for which $C = 0$ are of interest. Also, the least squares estimate of $x(T)$, now denoted by $\hat{x}(T)$ to emphasize the sequential nature of the problem, is $r(0, T)$. The result is

$$\frac{dP}{dt} C + \frac{d\hat{x}}{dT} - P(T) \left[\frac{\partial H^*}{\partial r} (T, \hat{x}, C) + \right. \qquad (F.17)$$

$$\left. \frac{\partial^2 H^*}{\partial r^2} (T, \hat{x}, C) \, PC \right] = \frac{\partial H^*}{\partial C} (T, \hat{x}, C)$$

$$+ \frac{\partial^2 H^*}{\partial r \partial C} (T, \hat{x}, C) \, PC$$

258

where

$$\frac{\partial^2 H^*}{\partial r^2} (T, \hat{x}, C) = \left\{ \frac{\partial^2 H^*}{\partial r_i \partial r_j} (T, \hat{x}, C) \right\}$$

(F.18)

$$\frac{\partial^2 H^*}{\partial r \partial C} (T, \hat{x}, C) = \left\{ \frac{\partial^2 H^*}{\partial r_j \partial C_i} (T, \hat{x}, C) \right\}$$

At this point it will be convenient to write the equations in component form. The summation convention will be used, i.e. if an index is repeated in a term then summation is implied. For example

$$a_{ij} b_{mj} C_m = \sum_m \sum_j a_{ij} b_{mj} C_m$$

(F.19)

Also G_{ij} or $(G)_{ij}$ will mean the (i,j)-th element of the matrix G while C_j will denote the j-th component of the vector C.

Writing equation (F.17) in component form gives

$$\frac{dP_{ij}}{dt} C_j + \frac{d\hat{x}_i}{dt} - P_{ij} \left[\frac{\partial H^*}{\partial r_j} \bigg|_{r=\hat{x}} + \left(\frac{\partial^2 H^*}{\partial r^2} \right)_{j\ell} \bigg|_{r=\hat{x}} (PC)_\ell \right]$$

(F.20)

$$= \frac{\partial H^*}{\partial C_i} (T, \hat{x}, C) + \left(\frac{\partial^2 H^*}{\partial r \partial C} \right)_{ij} \bigg|_{r=\hat{x}} (PC)_j$$

It is now necessary to digress in order to determine the various partials of H* as given by equation (F.10)

a)
$$\frac{\partial H^*}{\partial r_j}(T, r, C)\Big|_{r=\hat{x}} = -2\Big[h'_{\hat{x}} Q(y-h)\Big]_j \tag{F.21}$$

$$+ (g_{\hat{x}})_{\ell j} C_\ell - \frac{1}{4} C_\ell \frac{\partial}{\partial \hat{x}_j} (k V^{-1} k')_{\ell m} C_m$$

b)
$$\left\{\frac{\partial^2 H^*}{\partial r^2}(T, \hat{x}, C)\right\}_{j\ell} = \frac{\partial}{\partial r_\ell}\left(\frac{\partial H^*}{\partial r_j}\right)\Big|_{r=\hat{x}} \tag{F.22}$$

$$= -2 \frac{\partial}{\partial \hat{x}_\ell}\Big[h'_{\hat{x}} Q(y-h)\Big]_j$$

$$+ \frac{\partial}{\partial \hat{x}_\ell}(g_{\hat{x}})_{jm} C_m - \frac{1}{4} C_n \frac{\partial^2}{\partial \hat{x}_\ell \partial \hat{x}_j}(k V^{-1} k')_{nm} C_m$$

c)
$$\frac{\partial H^*}{\partial C_j}(T, \hat{x}, C) = g_i(t, \hat{x}) - \frac{1}{2}(k V^{-1} k')_{i\ell} C_\ell$$

$$\tag{F.23}$$

d)
$$\left\{\frac{\partial^2 H^*}{\partial r \partial C}(T, \hat{x}, C)\right\}_{ij} = \frac{\partial}{\partial \hat{x}_j}\left(\frac{\partial H^*}{\partial C_i}\right) \tag{F.24}$$

$$= (g_{\hat{x}})_{ij} - \frac{1}{2}\frac{\partial}{\partial \hat{x}_j}(k V^{-1} k')_{i\ell} C_\ell$$

Substituting equations (F.21, .22, .23, .24) into equation (F.20) gives

$$\frac{dP_{ij}}{dT} C_j + \frac{d\hat{x}_i}{dT} - P_{ij} \left\{ -2\left[h'_{\hat{x}} Q(y - h)\right]_j + (g_{\hat{x}})_{\ell j} C_\ell \right.$$

(F.25)

$$- \frac{1}{4} C_\ell \frac{\partial}{\partial \hat{x}_j} (k V^{-1} k')_{\ell m} C_m + \left\{ -2 \frac{\partial}{\partial \hat{x}_\ell} \left[h'_{\hat{x}} Q(y-h)\right]_j \right.$$

$$\left. + \frac{\partial}{\partial \hat{x}_\ell} (g_{\hat{x}})_{jm} C_m - \frac{1}{4} C_n \frac{\partial^2}{\partial \hat{x}_\ell \partial \hat{x}_j} (k V^{-1} k')_{nm} C_m \right\} (PC)_\ell \right\}$$

$$= g_i(t, \hat{x}) - \frac{1}{2} (k V^{-1} k')_{i\ell} C_\ell + (g\hat{x})_{ij} \quad (PC)_j$$

$$- \frac{1}{2} \frac{\partial}{\partial \hat{x}_j} (k V^{-1} k')_{i\ell} C_\ell (PC)_j$$

Collecting terms of order C^0 and those of order C^1 and higher yields the following:

Terms of order C^0

$$\frac{d\hat{x}_i}{dT} - P_{ij} \left\{ -2\left[h'_{\hat{x}} Q(y - h)\right]_j \right\} = g_i(t, \hat{x}) \qquad \text{(F.26)}$$

Rewriting equation (F.26)

$$\frac{d\hat{x}_i}{dT} = g_i(t, \hat{x}) - 2P_{ij}\left[h'_{\hat{x}} Q(y-h)\right]_j \qquad (F.27)$$

Terms of order C^1 and higher

$$\frac{dP_{i\ell}}{dT} C_\ell - P_{ij}\left\{(g_{\hat{x}})_{\ell j} C_\ell - 2\frac{\partial}{\partial\hat{x}_\ell}\left[h'_{\hat{x}} Q(y-h)\right]_j (PC)_\ell\right\}$$

$$\qquad (F.28)$$

$$= -\frac{1}{2}(k\ V^{-1}\ k')_{i\ell} C_\ell + (g_{\hat{x}})_{i\ell} (PC)_\ell$$

$$+ \text{(terms of order } C^2 \text{ and } C^3)$$

Rewriting equation (F.28)

$$\frac{dP_{i\ell}}{dT} C_\ell = P_{ij}(g_{\hat{x}})_{\ell j} C_\ell + (g_{\hat{x}})_{i\ell} P_{\ell m} C_m$$

$$\qquad (F.29)$$

$$-\frac{1}{2}(k\ V^{-1}\ k')_{i\ell} C_\ell$$

$$- 2P_{ij}\frac{\partial}{\partial\hat{x}_m}\left[h'_{\hat{x}} Q(y-h)\right]_j P_{m\ell}\ C_\ell$$

$$+ \text{(terms of order } C^2 \text{ and } C^3)$$

If equations (F.27) and (F.29) are satisfied then so is equation (F.25). The solution of equations (F.25) of interest are those for which C = 0. Hence the estimator equations become

$$\frac{d\hat{x}_i}{dT} = g_i(T, \hat{x}) - 2P_{ij}\left[h_{\hat{x}}' \, Q(y-h)\right]_j$$

<div align="right">(F.30)</div>

$$\frac{dP_{ij}}{dT} = P_{i\ell}(g_{\hat{x}})_{j\ell} + (g_{\hat{x}})_{i\ell} \, P_{\ell j}$$

$$- \frac{1}{2} (k \, V^{-1} \, k')_{ij}$$

$$- 2P_{i\ell} \, \frac{\partial}{\partial \hat{x}_m} \left[h_{\hat{x}}' \, Q(y-h)\right]_\ell \, P_{mj}$$

Substituting $-P_{ij}$ for P_{ij} then in vector, matrix notation equations (F.30) become

$$\frac{d\hat{x}}{dT} = g(T, \hat{x}) + 2P(T) \, H(t, \hat{x}) \, Q\{y - h(t, \hat{x})\}$$

<div align="right">(F.31)</div>

$$\frac{dP}{dT} = g_{\hat{x}}(T, \hat{x})P + Pg_{\hat{x}}'(T, \hat{x}) + 2P\left[H \, Q\{y - h(t, \hat{x})\}\right]_{\hat{x}} P$$

$$+ \frac{1}{2} k(T, \hat{x}) \, V^{-1}(T, \hat{x}) \, k'(T, \hat{x})$$

where

$$H(t, \hat{x}) = \left\{ \frac{\partial h_i}{\partial \hat{x}_j} \right\}'$$

$$\left[H \, Q \, \{y - h(t, \hat{x})\} \right]_{\hat{x}} \text{ is an } n \times n \text{ matrix with ith}$$

Column $\frac{\partial}{\partial \hat{x}_i} \left[HQ \, \{y - h(t, \hat{x})\} \right]$